

The microtype package

Subliminal refinements towards typographical perfection

R Schlicht
w.m.l@gmx.net

v2.5a
2013/05/23

The microtype package provides a \LaTeX interface to the micro-typographic extensions that were introduced by pdf \TeX and have since also propagated to X \TeX and Lua \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 © 2004–2013 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	12
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	29
14	Implementation	33
14.1	Preliminaries	34
	Debugging [35] Requirements [37] Declarations [40] Auxiliary macros [41] Compatibility [49]	
14.2	Font setup	54
	Protrusion [59] Expansion [66] Interword spacing (glue) [68] Additional kern- ing [70] Tracking [71] Disabling ligatures [81] Loading the configuration [82] Translating characters into slots [86] Hook into L ^A T _E X's font selection [93] Context-sensitive setup [96]	
14.3	Configuration	98
	Font sets [98] Variants and aliases [104] Disabling ligatures [105] Interaction with babel [105] Fine tuning [105] Character inheritance [112] Permuta- tion [114]	

14.4	Package options	117
	Declaring the options [117] Loading the definition file [121] Reading the configuration file [121] Hook for other packages [122] Changing options later [123] Processing the options [125]	
15	Configuration files	134
15.1	Font sets	134
15.2	Font variants and aliases	135
15.3	Interaction with babel	136
15.4	Note on admissible characters	137
15.5	Character inheritance	137
	OT1 [137] T1 [138] LY1 [138] OT4 [139] QX [139] T5 [140] EU1, EU2 [141] Euro symbols [141]	
15.6	Tracking	142
15.7	Font expansion	142
15.8	Character protrusion	144
	Normal [145] Italics [153] Small caps [164] Italic small caps [167] Text companion [168] Computer Modern math [173] AMS symbols [177] Euler [180] Euro symbols [184]	
15.9	Interword spacing	185
	Nonfrenchspacing [187]	
15.10	Additional kerning	188
	French [189] Turkish [189]	
16	OpenType configuration files	190
16.1	Character inheritance	190
16.2	Character protrusion	195
17	Auxiliary file for micro fine tuning	210
A	The title logo	212
A.1	Macros	212
A.2	Document	217
B	The letterspacing illustration	217
B.1	Macros	217
B.2	Document	220
C	Change history	222
D	Index	230
E	The L^AT_EX Project Public License	238

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	84

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 characters of a font is especially useful for languages whose typographical tradition requires certain characters to be separated by a space. For example, it is customary in French typography to add a small space before question mark, exclamation mark and semi-colon, and a bigger space before the colon and the guillemets. Until now, this could only be achieved by making

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

these characters active (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 `mimetype` are but a first approximation, and I would highly welcome corrections and improvements. I suggest reading the reasoning behind the settings in section 15.9.

The possibility, finally, to *disable all ligatures* of a font may be useful for typewriter fonts.

The `mimetype` 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{mimetype}
```

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 respective micro-typographic feature, either via the respective package option or with the `\mimetypesetup` command (section 3).
- Select the fonts to which this feature should be applied by declaring and activating ‘sets of fonts’. Some 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 \LaTeX 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 \TeX 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 `pdf \TeX` versions older than 1.20, in DVI output mode (see section 3.5), or with `X \TeX` . 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 `pdf \TeX`):

```
\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 \TeX` (you may use the `LetterSpace` option of the `fontspec` package instead). With `Lua \TeX` , you need to load the fonts with the `fontspec` option `Renderer=Basic`. See the `fontspec` manual for details.

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 \TeX` or `Lua \TeX` .

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	★	★	★	☒	☒	☒ ^a
LuaT _E X	≥ 0.30	DVI	★	☒	∅	∅	∅	∅
		PDF	★	★	★	∅	∅	∅
	≥ 0.62	DVI	★	☒	∅	∅	∅	∅
		PDF	★	★	★	∅	∅	☒
X _Y L _A T _E X	≥ 0.9997	PDF	★	∅	∅	∅	∅	

★ = enabled ☒ = not enabled ∅ = not available ^a ≥ 1.40.4 recommended

In table 1, you find 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_YL_AT_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 version pdfT_EX 1.20 (and with LuaT_EX), the expanded instances of the fonts may be calculated automatically and at run-time instead of the user having to prepare the instances in advance. This option is true by default provided that you are using a T_EX engine with this capability and the output mode is PDF;

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

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

stretch $\langle integer \rangle$ 20

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

step $\langle integer \rangle$ * 1

Fonts are not expanded by arbitrary amounts but only by certain discrete steps within the expansion limits. With recent versions of pdfTeX (1.40 or newer) or LuaTeX, this option is by default set to 1, in order to allow trying the maximum number of font instances, and hence to guarantee the best possible output.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 ‘T’). This is called *selected expansion*, and its usage allows increasing the stretch and shrink limits (to, say, 30 instead of 20); however, the gain is limited since at the same time the average stretch variance will be decreased. Therefore, this option is by default set to `false`, so that all characters will be expanded by the same amount. See section 5.2 for a more detailed discussion.

3.4 Tracking

pdfTeX 1.40 | LuaTeX 0.62

letterspace $\langle integer \rangle$ 100

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

3.5 Miscellaneous options

DVIoutput true, false * false

pdfTeX and LuaTeX are not only able to generate PDF output but can also spit out DVI files.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. 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.

<code>draft</code>	true, false	false
<code>final</code>	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.	
<code>verbose</code>	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> .	
<code>babel</code>	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.	
<code>config</code>	<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=mycrottype</code> .	

3.6 Changing options later

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

Inside the preamble, this command accepts all package options described above (except for `config`). In the document body, this command may be used to change the general settings of the micro-typographic extensions. It then accepts all options from section 3.1: `expansion`, `protrusion` and `activate`, which in turn may receive the values true, false, `compatibility` or `nocompatibility`, and `tracking`, `kerning` 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. This package defines a font set called ‘basic`text`’ in the main configuration file as follows:

```
\DeclareMicrotypeSet{basictext}
{ encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2},
  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 ‘all`text`’, which is declared as:

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

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	∅	∅	∅	∅	∅
alltext (allmath)	Text encodings, TS1 (OML, OMS, U)	∅	∅	∅	∅
basictext (basicmath)	Text encodings (OML, OMS)	$\rm*$, $\sf*$	$\md*$	∅	\normalsize , \footnotesize , \small , \large
smallcaps	Text encodings	∅	∅	$\sc*$	∅
footnotesize	Text encodings, TS1	∅	∅	∅	$-\small$
scriptsize	Text encodings, TS1	∅	∅	∅	$-\footnotesize$
normalfont	$\encoding*$	$\family*$	$\series*$	$\shape*$	\normalsize

'Text encodings' = OT1, T1, T2A, LY1, OT4, QX, T5, EU1, EU2 '\...*' = '\...default'

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

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

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

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

Table 2 lists the nine 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 [$\langle features \rangle$] { $\langle set name \rangle$ }

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.

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

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 $\langle \text{character} \rangle = \langle \text{protrusion factors} \rangle$ pairs.

The characters may be specified either as a single character (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 (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 `\A` are valid, provided the character is actually declared in both the input and the font encoding. With LuaTeX or XeTeX, you may additionally specify a (font-specific) glyph name, prefixed with `'` (e.g., the ‘fl’ ligature as `/f_1`). Note that you also have the possibility to declare lists of characters that should inherit settings (see section 5.6).

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 can omit either number if the character should not be protruded on that side, but must not drop the separating comma.

Options:

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

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

Thus, the configuration may be simplified considerably. You can for instance create a default list for a font; settings for other shapes or series can then load these settings, and extend or overwrite them (since the value that comes last will take precedence). Font settings will be loaded recursively. The following options will affect all loaded lists, 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 (`={\left},\right\}`), 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 that all characters of a particular font (*set*) should be expanded or shrunk by the same amount, you would have to declare an empty list for these fonts.

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 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

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.

stretch, shrink and step, their respective global value will be used (that is, no calculation will take place).

As a practical example, suppose you have a paragraph containing a widow that could easily be avoided by shrinking the font a little bit more. 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 an `unnecessary' widow.}
```

This method of employing contexts to temporarily apply different expansion parameters only works with pdfTeX version 1.40.4 or later (for older versions, a dirty trick is laid out in section 14.2 on page 56). 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 TeX for a long time – is the adjustment of tracking, i.e., the uniform addition or subtraction of letter space to/from all the characters in a font. For example, it is good typographic practice to slightly 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.⁶ The `\SetTracking` command allows specifying

⁶ 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.

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 1 em (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 1 em.

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 1 em (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 1 em (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 As far as pdf \TeX is concerned, ligatures in letterspaced fonts would 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. This is not recommended, however, since it also entails that kerning will be switched off.⁷ The default settings disable ligatures for the character 'f' only, i.e., 'ff', '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 \lslig (see section 7).

⁷ The inseparable connexion of ligatures and kerns is a limitation of \TeX that will not be lifted before the advent of Lua \TeX .

⁸ With pdf \TeX 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.

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 I would in no way recommend; they are only for 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 $50/1000\text{em}$, fonts smaller than `\small` by 0.02em , and to decrease the tracking of large type by 0.02em . You can achieve this 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 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.

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

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, TeX will insert a so called glue, which is characterised by three parameters – the normal distance between two words, the maximum amount of space that may be added to it, and the maximum amount that may be subtracted. The latter two parameters come into effect whenever TeX tries to break a paragraph into lines and does not succeed; it can then stretch or shrink the spaces between words. These three parameters are specific to each font.

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

The *spacing settings* are declared as pairs of $\langle character \rangle = \langle spacing factors \rangle$, where the latter consist of three numbers: first, the additional kern inserted after this character if it appears before an interword space, second, the additional stretch amount, and third, the additional shrink amount. All values may also be negative, in which case the dimensions will be decreased. Not all values have to be specified, however, the settings must 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 $\langle dimension \rangle$ and, additionally, **space**. The latter will measure the values in thousandths of the respective space dimension set by the font. By default, the unit is measured by the space dimensions. For example, with these (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 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 \AA , $\text{\acute{A}}$, $\text{\hat{A}}$, $\text{\tilde{A}}$, \AA , \AA and \AA 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 Lua \TeX and X \TeX , 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 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}
```

Table 3:

Fonts with tailored protrusion settings

Font family (NFSS code)	Features	
	Encodings	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, EU2 [Latin, Greek]	n, it, (sl) ^d
Charis SIL	EU1, EU2 [Latin, Cyrillic, Greek]	n, it, sc
Palatino Linotype ⁱ	EU1, EU2 [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 (mdbch), MicroPress's chmath (chr)
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, T_EX Gyre Pagella (qpl), FPL Neu (fp9x, fp9j)
h Aliases: txfonts (txr), qfonts/QuasiTimes, T_EX Gyre Termes (qtm)
i Alias: T_EX Gyre Pagella, Palatino LT Std, Palatino
j Aliases: Latin Modern (lmsy, lmm)
k Alias: eulervm (zeur, zeus)

`\DeclareMicrotypeAlias` {*font name*} {*alias font*}

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

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

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

`\LoadMicrotypeFile` {*font name*}

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

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.

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.¹⁰

```
\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`, `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 LATEX 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
```

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:

¹⁰ With Lua_T_EX, you have to load the fonts with the `fontspec` option 'Renderer=Basic'.

```
\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.¹¹ 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.

```
\lslig {⟨ligature⟩}
```

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

¹¹ 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.

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 to solve 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’, ligatures shown in red, inhibited ligatures in green).

```
\SetTracking[no ligatures={f}]{encoding = LY, family = yfrak}{100}
\textfrak{\lsstyle Aus:s{\kern0pt}ichts:los{\kern0pt}igkeit}
```

```
\SetTracking[no ligatures={f,s,c}]{encoding = LY, family = yfrak}{100}
\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 \LaTeX , 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

pdfTeX 1.30 | LuaTeX 0.30

`\DisableLigatures` [*characters*] {*set of fonts*}

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

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

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.¹²

¹² With LuaTeX, you have to load the fonts with the `fontspec` option ‘`Renderer=Basic`’.

9 Hints and caveats

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

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

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

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

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

You might want to disable protrusion in verbatim environments. As you know by now, `microtype` will by default activate character protrusion for all fonts contained in the font set `'alltext'`. This also includes the typewriter font. Although it does make sense to protrude the typewriter font if it appears in running text (like, for example, in this manual), this is probably not desirable inside the `verbatim` environment. However, `microtype` has no knowledge about the context that a font appears in but will solely decide by examining its attributes. Therefore, you have to take care of disabling protrusion in `verbatim` environments for yourself (that is, if you don't want to disable protrusion for the typewriter font altogether, by choosing a different font set). While the `\microtypesetup` command has of course been designed for cases like this, you might find it tiresome to repeat it every time if you are using the `verbatim` environment frequently. The following line, added to the document's preamble, would serve the same purpose:

```
\g@addto@macro\@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 pdf \TeX 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:

- If you want to use 8-bit characters in the configuration, you have to load the `inputenc` package first. Unicode input is also supported (when loading `inputenc` with the `utf8` or the `utf8x` option, or out of the box with $X_{\text{}}\TeX$ and Lua \TeX). 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 Lua \TeX , 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.¹³

¹³ They make use of features provided by `luaotfload` (via `fontspec`).

- With LuaTeX, load fontspec before microtype.
- 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 the CJK package is font expansion.

Possible error messages and how to get rid of them:

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

Harald Harders has contributed protrusion settings for Adobe Minion. I would also like to thank him for a number of bug reports and suggestions he had to make. *Andreas 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 suggestion, especially concerning the new extensions of interword spacing adjustment and additional character kerning. *Georg Duffner* has patiently tested `microtype` under X \TeX and Lua \TeX with his beautiful OpenType font EB Garamond¹⁴. 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 Lua \TeX 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.

14 Available from CTAN at </fonts/ebgaramond>, including configuration files for `microtype`.

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

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

Hàn Thế Thành, ‘Font-specific issues in pdf \TeX ’, in: *TUGBoat*, vol. 29 (2008), no. 1: ‘EuroBach \TeX 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 pdf \TeX user manual*, 23 November 2010. (Available from CTAN at [/systems/pdftex/](http://systems/pdftex/); latest version at <http://sarovar.org/projects/pdftex/>)

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

\LaTeX 3 Project Team, *\LaTeX 2 ϵ font selection*, 27 November 2005. (Available from CTAN at [/macros/latex/doc/fntguide.pdf](http://macros/latex/doc/fntguide.pdf))

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

Melchior Franz, *The soul package*, 17 November 2003. (Available from CTAN at [/macros/latex/contrib/soul/](http://macros/latex/contrib/soul/)). See also Heiko Oberdiek’s extension of this package, *soulutf8*, which adds Unicode support. (Available from CTAN at [/macros/latex/contrib/oberdiek/](http://macros/latex/contrib/oberdiek/))

Will Robertson, Khaled Hosny, *The fontspec package*, 16 March 2013. (Available from CTAN at [/macros/latex/contrib/fontspec/](http://macros/latex/contrib/fontspec/))

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.

2.5 (2013/03/13)

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

1.8 (2005/06/23)

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

1.7 (2005/03/23)

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

1.6 (2005/01/24)

- New option ‘factor’ to influence protrusion resp. expansion of all characters of a font or font set [3.2, 5]
- When pdf \TeX 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 \LaTeX 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*\slig[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@n1#1{\MT@warning{#1\@gobble}}
\MT@info@nl    78 <*package>
\MT@info@nl    79 \def\MT@info{\PackageInfo\MT@MT}
\MT@vinfo      80 \def\MT@info@n1#1{\MT@info{#1\@gobble}}
\MT@error      81 \let\MT@vinfo\@gobble
\MT@error      82 \def\MT@error{\PackageError\MT@MT}
\MT@warn@err   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@info      0: almost none
\MT@info@nl   1: + sets & lists
              2: + heirs
              3: + slots
              4: + factors
```

```
87 <*debug>
88 \MT@warning@n1{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 . . .

```
\tracingmicrotypeinpdf=2
```

`\MT@pdf@annot` 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 `\MT@addto@annot` and `\ifMT@inannot`. `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 `\tracingmicrotypeinpdfallfalse`, 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: \LaTeX

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

```

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

```

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

```

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

```

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

```

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

```

We check whether we are running pdf \TeX , X \TeX , or Lua \TeX , 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 \TeX Live 2004 hack which undefines the pdf \TeX primitives in the format in order to hide the fact that pdf \TeX is being run from the

user. This has been *fixed* in T_EX Live 2005.

```
168 \ifx\normalpdftexversion\@undefined \else
169   \let\pdftexversion\normalpdftexversion
170   \let\pdftexrevision\normalpdftexrevision
171   \let\pdfoutput\normalpdfoutput
172 \fi
```

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

```
\MT@engine@toold
173 \let\MT@engine\relax
174 <letterspace>\def\MT@engine@toold{0}
175 \ifx\pdftexversion\@undefined \else
176   \ifx\pdftexversion\relax \else
177     \def\MT@engine{pdf}
178     <letterspace> \let\MT@pdf@or@lua\@firstoftwo
179     <letterspace> \ifnum\pdftexversion > 139 \def\MT@engine@toold{1}\fi
180     \ifx\directlua\@undefined \else
181       \ifx\directlua\relax \else
182         \def\MT@engine{lua}
183         <letterspace> \let\MT@pdf@or@lua\@secondoftwo
184         <letterspace> \ifnum\luatexversion < 62 \def\MT@engine@toold{0}\fi
185       \fi
186     \fi
187   \fi
188 \fi
189 <*package>
190 \ifx\MT@engine\relax
191   \ifx\XeTeXversion\@undefined \else
192     \ifx\XeTeXversion\relax \else
193       \def\MT@engine{xe}
194     \fi
195   \fi
196 \fi
197 </package>
198 </package|letterspace>
```

`\MT@pdftex@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@pdftex@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 1em (≥ 0.14h)
- 4: + automatic font expansion; protrusion no longer has to be set up first; scale factor fixed to 1000; default `\efcode` = 1000 (≥ 1.20)
- 5: + `\(left,right)marginkern`; `\pdfnoligatures`; `\pdfstrcmp`; `\pdfescapestring` (≥ 1.30)
- 6: + adjustment of interword spacing; extra kerning; `\letterspacefont`; `\pdfmatch`¹⁵; `\pdftracingfonts`; always e-T_EX (≥ 1.40)
- 7: + `\letterspacefont` doesn't disable ligatures and kerns; `\pdfcopyfont` (≥ 1.40.4)

```
199 <*pdftex-def>
200 <debug>\MT@info@n{0}{this is pdftex \the\pdftexversion(\pdftexrevision)}
201 \def\MT@pdftex@no{7}
```

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

```

202 \ifnum\pdfTeXversion = 140
203   \ifnum\pdfTeXrevision < 4
204     \def\MT@pdfTeX@no{6}
205   \fi
206 \else
207   \ifnum\pdfTeXversion < 140
208     \def\MT@pdfTeX@no{5}
209     \ifnum\pdfTeXversion < 130
210       \def\MT@pdfTeX@no{4}
211       \ifnum\pdfTeXversion < 120
212         \def\MT@pdfTeX@no{3}
213         \ifnum\pdfTeXversion = 14
214           \ifnum \expandafter`\pdfTeXrevision < `h
215             \def\MT@pdfTeX@no{2}
216           \ifnum \expandafter`\pdfTeXrevision < `f
217             \def\MT@pdfTeX@no{1}
218           \fi
219         \fi
220       \else
221         \ifnum\pdfTeXversion < 14
222           \def\MT@pdfTeX@no{1}
223         \fi
224       \fi
225     \fi
226   \fi
227 \fi
228 \fi
229 <debug>\MT@info@n1{0}{pdfTeX no.: \MT@pdfTeX@no}
230 </pdfTeX-def>

```

\MT@xetex@no Xe_{La}TeX supports character protrusion since version 0.9997.

```

231 <*xetex-def>
232 <debug>\MT@info@n1{0}{this is xetex (\the\XeTeXversion\XeTeXrevision)}
233 \ifdim 0\XeTeXrevision pt < 0.9997pt
234   \def\MT@xetex@no{1}
235 \else
236   \def\MT@xetex@no{2}
237 \fi
238 <debug>\MT@info@n1{0}{xetex no.: \MT@xetex@no}
239 </xetex-def>

```

\MT@luatex@no Cases for Lua_{La}TeX (\luatexversion ought to have been enabled by the format):

0: N/A

1: Lua_{La}TeX (< 0.36)

2: + \directlua without state number (≥ 0.36)

3: + \letterspacefont (≥ 0.62).

```

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

```

\MT@lua Communicate with lua. Beginning with Lua_{La}TeX 0.36, \directlua no longer requires a state number.

```

242 \def\MT@lua{\directlua}
243 \def\MT@luatex@no{3}
244 \ifnum\luatexversion<62
245   \def\MT@luatex@no{2}
246   \ifnum\luatexversion<36
247     \def\MT@lua{\directlua0}
248   \def\MT@luatex@no{1}
249 \fi
250 \fi

```

```

251 <debug>\MT@info@n1{0}{luatex no.: \MT@luatex@no}
252 </luatex-def>

253 <*&pdfTEX-def|xetex-def|letterspace>
254 \ifnum
255 <pdfTEX-def|xetex-def> \cSname MT@\MT@engine tex@no\endcsname < 2
256 <letterspace> \MT@engine@toold=\z@
257 \MT@warning@n1{You
258 <*&letterspace>
259 \ifx\MT@engine\relax
260 don't seem to be using pdfTEX or luatex.\MessageBreak
261 Try running `pdfTEX' or `luatex' instead of.\MessageBreak
262 \ifx\XeTeXversion\undefined\else xe\fi tex'%
263 \else
264 </letterspace>
265 are using a \MT@engine tex version older than
266 <pdfTEX-def> 0.14f%
267 <xetex-def> 0.9997%
268 <letterspace> \MT@pdf@or@lua{1.40}{0.62}%
269 .\MessageBreak
270 ~\MT@MT' does not work with this version.\MessageBreak
271 Please install a newer version of \MT@engine tex%
272 <letterspace> \fi
273 .\MessageBreak I will quit now}
274 \MT@clear@options
275 \endinput\fi
276 </pdfTEX-def|xetex-def|letterspace>

```

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

```

277 <*&package|letterspace>
278 \RequirePackage{keyval}[1997/11/10]
279 <*&package>

```

\MT@toks We need a token register.

```
280 \newtoks\MT@toks
```

\ifMT@if@ A scratch if.

```
281 \newif\ifMT@if@
```

14.1.3 Declarations

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

```

\ifMT@expansion 282 \newif\ifMT@protrusion
\ifMT@auto 283 \newif\ifMT@expansion
\ifMT@selected 284 \newif\ifMT@auto
285 \newif\ifMT@selected
\ifMT@noligatures 286 \newif\ifMT@noligatures
\ifMT@draft 287 \newif\ifMT@draft
\ifMT@spacing 288 \newif\ifMT@spacing
\ifMT@kerning 289 \newif\ifMT@kerning
\ifMT@tracking 290 \newif\ifMT@tracking
\ifMT@babel 291 \newif\ifMT@babel

```

\MT@MF@babel ... and numbers.

```

\MT@ex@level 292 \let\MT@pr@level\tw@
\MT@pr@factor 293 \let\MT@ex@level\tw@
294 \let\MT@pr@factor@m
\MT@ex@factor 295 \let\MT@ex@factor@m
\MT@sp@factor 296 \let\MT@sp@factor@m
\MT@kn@factor 297 \let\MT@kn@factor@m

```


<code>\MT@pr@unit</code>	Default unit for protrusion settings is character width, for spacing space, for kerning
<code>\MT@sp@unit</code>	(and tracking) 1 em.
<code>\MT@kn@unit</code>	298 <code>\let\MT@pr@unit\@empty</code> 299 <code>\let\MT@sp@unit\m@ne</code> 300 <code>\def\MT@kn@unit{1em}</code>
<code>\MT@stretch</code>	Expansion settings.
<code>\MT@shrink</code>	301 <code>\let\MT@stretch\m@ne</code>
<code>\MT@step</code>	302 <code>\let\MT@shrink \m@ne</code> 303 <code>\let\MT@step \m@ne</code>
<code>\MT@pr@min</code>	Minimum and maximum values allowed by pdf \TeX .
<code>\MT@pr@max</code>	304 <code>\def\MT@pr@min{-\@m}</code>
<code>\MT@ex@min</code>	305 <code>\let\MT@pr@max\@m</code>
<code>\MT@ex@max</code>	306 <code>\let\MT@ex@min\z@</code> 307 <code>\let\MT@ex@max\@m</code>
<code>\MT@sp@min</code>	308 <code>\def\MT@sp@min{-\@m}</code>
<code>\MT@sp@max</code>	309 <code>\let\MT@sp@max\@m</code>
<code>\MT@kn@min</code>	310 <code>\def\MT@kn@min{-\@m}</code>
<code>\MT@kn@max</code>	311 <code>\let\MT@kn@max\@m</code> 312 <code>\langle/package\rangle</code>
<code>\MT@tr@min</code>	313 <code>\def\MT@tr@min{-\@m}</code>
<code>\MT@tr@max</code>	314 <code>\let\MT@tr@max\@m</code> 315 <code>\langle/package\rangle</code>
<code>\MT@factor@default</code>	Default factor. 316 <code>\def\MT@factor@default{1000 }</code>
<code>\MT@stretch@default</code>	Default values for expansion.
<code>\MT@shrink@default</code>	317 <code>\def\MT@stretch@default{20 }</code> 318 <code>\def\MT@shrink@default{20 }</code>
<code>\MT@letterspace</code>	Default value for letterspacing (in thousandths of 1 em).
<code>\MT@letterspace@default</code>	319 <code>\langle/package\rangle</code> 320 <code>\let\MT@letterspace\m@ne</code> 321 <code>\def\MT@letterspace@default{100}</code> 322 <code>\langle/package\rangle</code>
<code>\ifMT@document</code>	Our private test whether we're still in the preamble. 323 <code>\newif\ifMT@document</code> 324 <code>\langle/package\rangle</code> 325 <code>\langle/package letterspace\rangle</code>

14.1.4 Auxiliary macros

<code>\MT@requires@pdftex</code>	For definitions that depend on a particular pdf \TeX resp. Lua \TeX version.
<code>\MT@requires@luatex</code>	326 <code>\langle/pdftex-def luatex-def\rangle</code> 327 <code>\def</code> 328 <code>\langle/pdftex-def\rangle \MT@requires@pdftex%</code> 329 <code>\langle/luatex-def\rangle \MT@requires@luatex%</code> 330 <code>#1{\ifnum</code> 331 <code>\langle/pdftex-def\rangle \MT@pdftex@no</code> 332 <code>\langle/luatex-def\rangle \MT@luatex@no</code> 333 <code><#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}</code> 334 <code>\langle/debug+pdftex-def\rangle\MT@requires@pdftex6{</code> 335 <code>\langle/debug\rangle\pdftracingfonts=1</code> 336 <code>\langle/debug+pdftex-def\rangle}\relax</code> 337 <code>\langle/pdftex-def luatex-def\rangle</code>

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

```

338 {*luatex-def}
339 \RequirePackage{luatexbase}
340 \RequireLuaModule{microtype}
341 {/luatex-def}

```

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

```

342 {*luafile}
343 microtype = microtype or {}
344 local microtype = microtype
345
346 local microtype_module = {
347   name       = "microtype",
348   version    = 2.5,
349   date       = "2013/05/15",
350   description = "microtype module.",
351   author     = "E. Roux, R. Schlicht and P. Gesang",
352   copyright  = "E. Roux, R. Schlicht and P. Gesang",
353   license    = "LPPL",
354 }
355
356 if luatexbase and luatexbase.provides_module then
357   luatexbase.provides_module(microtype_module)
358 end
359
360 local find = string.find
361 local tex_write = tex.write
362
363 {/luafile}

```

To be continued, but first back to primitives.

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

```

364 {*package|letterspace}
365 \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.

```

366 \def\MT@exp@cs#1#2{\expandafter#1\csname#2\endcsname}
367 {*package}
368 \def\MT@exp@gcs#1#2{\begingroup\expandafter\endgroup\expandafter#1\csname#2\endcsname}

```

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

```

369 \def\MT@def@n{\MT@exp@cs\def}
370 \def\MT@gdef@n{\MT@exp@gcs\gdef}

```

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

```

371 {/package}
372 \def\MT@edef@n{\MT@exp@cs\edef}
373 {*package}
374 \def\MT@xdef@n{\MT@exp@gcs\xdef}

```

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

```

375 \def\MT@let@nc{\MT@exp@cs\let}
376 \def\MT@glet@nc{\MT@exp@gcs\MT@glet}

```

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

```

377 \def\MT@let@cn#1#2{\expandafter\let\expandafter#1\csname #2\endcsname}

```

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

```

378 \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
379 \def\MT@glet@nn{\MT@exp@gcs{\global\expandafter\MT@let@cn}}

```

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

```

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

```

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

```
381 /package
382 \def\MT@exp@one@n#1#2{\expandafter#1\expandafter{#2}}
```

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

```
383 \def\MT@exp@two@c#1{\expandafter\expandafter\expandafter#1\expandafter}
384 *package
```

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

```
385 \def\MT@exp@two@n#1#2#3{%
386   \expandafter\expandafter\expandafter
387   #1\expandafter\expandafter\expandafter
388   {\expandafter#2\expandafter}\expandafter{#3}}
```

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

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

```
\MT@ifdefined@c@TF
\MT@ifdefined@n@T
\MT@ifdefined@n@TF 389 \def\MT@ifdefined@c@T#1{%
390   ^^X \ifdefined#1\expandafter\@firstofone\else\expandafter\@gobble\fi
391   ^^Q \ifx#1\@undefined\expandafter\@gobble\else\expandafter\@firstofone\fi
392 }
393 /package
394 \def\MT@ifdefined@c@TF#1{%
395   ^^X \ifdefined#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
396   package^^Q \ifx#1\@undefined
397   package^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
398 }
399 \def\MT@ifdefined@n@T#1{%
400   ^^X \ifcsname#1\endcsname\expandafter\@firstofone\else\expandafter\@gobble\fi
401   package^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
402   package^^Q \expandafter\@gobble\else\expandafter\@firstofone\fi
403 }
404 \def\MT@ifdefined@n@TF#1{%
405   ^^X \ifcsname#1\endcsname\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
406   package^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
407   package^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
408 }
409 *package
```

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

```
\MT@detokenize@c
\MT@rem@last@space 410 \def\MT@detokenize@n#1{%
411   ^^X \expandafter\MT@rem@last@space\detokenize{#1} \@nil
412   ^^Q \string#1%
413 }
414 \def\MT@detokenize@c#1{%
415   ^^X \MT@exp@one@n\MT@detokenize@n#1%
416   ^^Q \MT@exp@two@c\MT@rem@last@space\strip@prefix\meaning#1 \@nil
417 }
418 \def\MT@rem@last@space#1 #2{#1%
419   \ifx\@nil#2\else \space
420   \expandafter\MT@rem@last@space\expandafter#2\fi
421 }
```

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

```
422 /package
423 \begingroup
424 \catcode`\%=12
425 \catcode`\&=14
426 \gdef\MT@ifempty#1{&
427   \if %#1&
```

```

428   \expandafter\@firstoftwo
429   \else
430   \expandafter\@secondoftwo
431   \fi
432 }
433 \endgroup
434 {*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).

435 {/package}
436 {/package|letterspace}
437 {pdftex-def}\MT@requires@pdftex6{
438 {letterspace}\MT@pdf@or@lua{
439 {*pdftex-def|letterspace}
440 \def\MT@ifint#1{%
441   \ifcase\pdfmatch{^-*[0-9]+ *$}{#1}\relax
442   \expandafter\@secondoftwo
443   \else
444   \expandafter\@firstoftwo
445   \fi
446 }
447 }{
448 {/pdftex-def|letterspace}
449 {*pdftex-def|xetex-def|letterspace}
450 \def\MT@ifint#1{%
451   \if!\ifnum9<1#1!\else?\fi
452   \expandafter\@firstoftwo
453   \else
454   \expandafter\@secondoftwo
455   \fi
456 }
457 {/pdftex-def|xetex-def|letterspace}
458 {pdftex-def|letterspace}}
459 {luatex-def}\def\MT@ifint#1{\csname\MT@lua{microtype.if_int}([\#1])\endcsname}
460 {*luafile}
461 local function if_int(s)
462   if find(s,"^-*[0-9]+ *$") then
463     tex_write("@firstoftwo")
464   else
465     tex_write("@secondoftwo")
466   end
467 end
468 microtype.if_int = if_int
469
470 {/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.)

471 {*pdftex-def}
472 \MT@requires@pdftex6{
473 \def\MT@ifdimen#1{%
474   \ifcase\pdfmatch{^[0-9]+([.][0-9]+)?|[.][0-9]+)%
475             (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
476   \expandafter\@secondoftwo
477   \else
478   \expandafter\@firstoftwo
479   \fi
480 }
481 }{
482 {/pdftex-def}
483 {*pdftex-def|xetex-def}
484 \def\MT@ifdimen#1{%

```

```

485 \setbox\z0=\hbox{%
486 \MT@count=1#1\relax
487 \ifnum\MT@count=\@ne
488 \aftergroup\@secondoftwo
489 \else
490 \aftergroup\@firstoftwo
491 \fi
492 }%
493 }
494 </pdfTeX-def>|xetex-def)
495 <pdfTeX-def>}
496 <luatex-def>\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen([[#1]])}\endcsname}
497 <luafile>
498 local function if_dimen(s)
499 if (find(s, "^-*[0-9]+(%a*) *$") or
500 find(s, "^-*[0-9]*[.][0-9]+(%a*) *$")) then
501 tex_write("@firstoftwo")
502 else
503 tex_write("@secondoftwo")
504 end
505 end
506 microtype.if_dimen = if_dimen
507
508 </luafile>
\MT@ifdim Test floating point numbers.
509 <package>
510 \def\MT@ifdim#1#2#3{%
511 \ifdim #1\p@ #2 #3\p@
512 \expandafter\@firstoftwo
513 \else
514 \expandafter\@secondoftwo
515 \fi
516 }
517 </package>
\MT@ifstreq Test whether two strings (fully expanded) are equal.
518 <pdfTeX-def>
519 \MT@requires@pdfTeX5{
520 \def\MT@ifstreq#1#2{%
521 \ifcase\pdfstrcmp{#1}{#2}\relax
522 \expandafter\@firstoftwo
523 \else
524 \expandafter\@secondoftwo
525 \fi
526 }
527 }{
528 </pdfTeX-def>
529 <pdfTeX-def>|xetex-def)
530 \def\MT@ifstreq#1#2{%
531 \edef\MT@res@a{#1}%
532 \edef\MT@res@b{#2}%
533 \ifx\MT@res@a\MT@res@b
534 \expandafter\@firstoftwo
535 \else
536 \expandafter\@secondoftwo
537 \fi
538 }
539 </pdfTeX-def>|xetex-def)
540 <pdfTeX-def>}
541 <luatex-def>\def\MT@ifstreq#1#2{\csname\MT@lua{microtype.if_str_eq([[#1]], [[#2]])}\endcsname}
542 <luafile>
543 local function if_str_eq(s1, s2)
544 if s1 == s2 then
545 tex_write("@firstoftwo")

```

```

546 else
547   tex_write("@secondoftwo")
548 end
549 end
550 microtype.if_str_eq = if_str_eq
551
552 /luafile

\MT@xadd    Add item to a list.
553 (*package)
554 \def\MT@xadd#1#2{%
555   \ifx#1\relax
556     \xdef#1{#2}%
557   \else
558     \xdef#1{#1#2}%
559   \fi
560 }

\MT@xaddb  Add item to the beginning.
561 \def\MT@xaddb#1#2{%
562   \ifx#1\relax
563     \xdef#1{#2}%
564   \else
565     \xdef#1{#2#1}%
566   \fi
567 }
568 /package

\MT@map@clist@n    Run <#2> on all elements of the comma list <#1>. This and the following is modelled
\MT@map@clist@c    after LATEX3 commands.
\MT@map@clist@    (*package | letterspace)
\MT@clist@function 570 \def\MT@map@clist@n#1#2{%
\MT@clist@break    571   \ifx\@empty#1\else
                    572     \def\MT@clist@function##1{#2}%
                    573     \MT@map@clist@#1,\@nil,\@nnil
                    574   \fi
                    575 }
                    576 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}
                    577 \def\MT@map@clist@#1,{%
                    578   \ifx\@nil#1%
                    579     \expandafter\MT@clist@break
                    580   \fi
                    581   \MT@clist@function{#1}%
                    582   \MT@map@clist@
                    583 }
                    584 \let\MT@clist@function\@gobble
                    585 \def\MT@clist@break#1\@nnil{}
                    586 (*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@    587 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break    588 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
                    589 \def\MT@map@tlist@#1#2{%
                    590   \ifx\@nnil#2\else
                    591     #1{#2}%
                    592     \expandafter\MT@map@tlist@
                    593     \expandafter#1%
                    594   \fi
                    595 }
                    596 \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@    597 \newif\ifMT@inlist@

```

```

598 \def\MT@in@clist#1#2{%
599   \def\MT@res@a#1,#1,#2##3\@nnil{%
600     \ifx##2\@empty
601       \MT@inlist@false
602     \else
603       \MT@inlist@true
604     \fi
605   }%
606   \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil
607 }

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

608 \def\MT@rem@from@clist#1#2{%
609   \def\MT@res@a#1,#1,#2\MT@res@a{##1,##2\MT@res@b}%
610   \def\MT@res@b#1,\MT@res@b##2\MT@res@b{\ifx,##1\@empty\else##1\fi}%
611   \xdef#2{\MT@exp@two@c\MT@res@b\MT@res@a\expandafter,#2,\MT@res@b,#1,\MT@res@a}%
612 }

\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.

613 \def\MT@in@tlist#1#2{%
614   \MT@inlist@false
615   \def\MT@res@a{#1}%
616   \MT@map@tlist@c#2\MT@in@tlist@
617 }
618 \def\MT@in@tlist@#1{%
619   \edef\MT@res@b{#1}%
620   \ifx\MT@res@a\MT@res@b
621     \MT@inlist@true
622   \expandafter\MT@tlist@break
623   \fi
624 }

\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@@       625 \def\MT@in@rlist#1{%
\MT@size@name         626   \MT@inlist@false
                      627   \MT@map@tlist@c#1\MT@in@rlist@
                      628 }
                      629 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
630 \def\MT@in@rlist@@#1#2#3{%
631   \MT@ifdim{#2}=\m@ne{%
632     \MT@ifdim{#1}=\MT@size
633     \MT@inlist@true
634     \relax
635   }%
636   \MT@ifdim\MT@size<{#1}\relax{%
637     \MT@ifdim\MT@size<{#2}%
638     \MT@inlist@true
639     \relax
640   }%
641 }%
642 \ifMT@inlist@
643   \def\MT@size@name{#3}%
644   \expandafter\MT@tlist@break
645   \fi
646 }

\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           647 </package>
                      648 \def\MT@loop#1\MT@repeat{%

```

```

649 \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
650 \MT@iterate \let\MT@iterate\relax
651 }
652 \let\MT@repeat\fi
\MT@while@num    Execute <#3> from <#1> up to (excluding) <#2> (much faster than LATEX's \@whilenum).
653 \def\MT@while@num#1#2#3{%
654   \@tempcnta#1\relax
655   \MT@loop #3%
656   \advance\@tempcnta \@ne
657   \ifnum\@tempcnta < #2\MT@repeat
658 }

\MT@do@font    Execute <#1> 256 times,
659 </package|letterspace>
660 <*pdftex-def|luatex-def|letterspace>
661 \def\MT@do@font{\MT@while@num\z@
662 <pdftex-def|letterspace> \cclvi
        resp. 1114111 times for LuaTEX (this is going to be slow, but LuaTEX is slow anyway
        – still, there ought to be a better way!)
663 <luatex-def> \MT@max@slot
664 }
665 </pdftex-def|luatex-def|letterspace>
        resp. for the whole font.
666 <*xetex-def>
667 \def\MT@do@font#1{%
668   \@tempcnta=\z@
669   \MT@loop #1%
670   \advance\@tempcnta \@ne
671   \ifnum\@tempcnta < \XeTeXcountglyphs\MT@font \MT@repeat
672 }
673 </xetex-def>
674 <*package>

\MT@count      Increment macro <#1> by one. Saves using up too many counters. The e-TEX way is
\MT@increment  slightly faster.
675 \newcount\MT@count
676 \def\MT@increment#1{%
677   ^^X \edef#1{\number\numexpr #1 + 1\relax}%
678   ^^Q \MT@count=#1\relax
679   ^^Q \advance\MT@count \@ne
680   ^^Q \edef#1{\number\MT@count}%
681 }

\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.
682 \def\MT@scale#1#2#3{%
683   ^^Q \multiply #1 #2\relax
684   \ifnum #3 = \z@
685     ^^X #1=\numexpr #1 * #2\relax
686   \else
687     ^^X #1=\numexpr #1 * #2 / #3\relax
688     ^^Q \divide #1 #3\relax
689   \fi
690 }

\MT@abbr@pr    Some abbreviations. Thus, we can have short command names but full-length log
\MT@abbr@ex    output.
\MT@abbr@pr@c  691 \def\MT@abbr@pr{protrusion}
\MT@abbr@ex@c
\MT@abbr@pr@inh
\MT@abbr@ex@inh
\MT@abbr@enl
\MT@abbr@sp
\MT@abbr@sp@c
\MT@abbr@sp@inh
\MT@abbr@kn

```



```

692 \def\MT@abbr@ex{expansion}
693 \def\MT@abbr@pr@c{protrusion codes}
694 \def\MT@abbr@ex@c{expansion codes}
695 \def\MT@abbr@pr@inh{protrusion inheritance}
696 \def\MT@abbr@ex@inh{expansion inheritance}
697 \def\MT@abbr@nl{noligatures}
698 \def\MT@abbr@sp{spacing}
699 \def\MT@abbr@sp@c{interword spacing codes}
700 \def\MT@abbr@sp@inh{interword spacing inheritance}
701 \def\MT@abbr@kn{kerning}
702 \def\MT@abbr@kn@c{kerning codes}
703 \def\MT@abbr@kn@inh{kerning inheritance}
704 \def\MT@abbr@tr{tracking}
705 \def\MT@abbr@tr@c{tracking amount}

```

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

```

\MT@rbba@expansion 706 \def\MT@rbba@protrusion{pr}
\MT@rbba@spacing    707 \def\MT@rbba@expansion{ex}
\MT@rbba@kerning    708 \def\MT@rbba@spacing{sp}
\MT@rbba@tracking   709 \def\MT@rbba@kerning{kn}
\MT@rbba@tracking   710 \def\MT@rbba@tracking{tr}

```

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

```

\MT@features@long 711 \def\MT@features{pr,ex,sp,kn,tr}
                   712 \def\MT@features@long{protrusion,expansion,spacing,kerning,tracking}

```

\MT@is@feature Whenever an optional argument accepts a list of features, we can use this command to check whether a feature exists in order to prevent a rather confusing ‘Missing \endcsname inserted’ error message. The feature (long form) must be in \@tempa, the type of list to ignore in (#1), then comes the action.

```

713 \def\MT@is@feature#1{%
714   \MT@exp@one@n\MT@in@clist\@tempa\MT@features@long
715   \ifMT@inlist@
716     \expandafter\@firstofone
717   \else
718     \MT@error{`\@tempa' is not an available micro-typographic\MessageBreak
719       feature. Ignoring #1}{Available features are: `~\MT@features@long'.}%
720     \expandafter\@gobble
721   \fi
722 }

```

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.

```

723 \@ifl@aded{tex}{wordcount}{%
724   \MT@warning@nl{Detected the `wordcount' utility.\MessageBreak
725     Disabling `~\MT@MT', since it wouldn't work}%
726   \MT@clear@options\endinput}\relax

```

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

don't have to be set up before microtype.

```
727 </package>
728 <*package|letterspace>
729 <plain>\MT@requires@latex1{
730 \let\MT@setup@\empty
```

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

```
731 \def\MT@addto@setup{\g@addto@macro\MT@setup@
```

Don't hesitate with `miniltx`.

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

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

```
733 \def\MT@with@package@T#1{\@ifpackageloaded{#1}\@firstofone\@gobble}
734 </package|letterspace>
735 <*package>
```

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

```
736 \def\MT@with@babel@and@T#1{%
737   \MT@ifdefined@n@T{opt@babel.\@pkgextension}{%
738     \expandtwoargs\MT@in@clist{#1}
739     {\csname opt@babel.\@pkgextension\endcsname,\@classoptionslist}%
740     \ifMT@inlist@expandafter\@gobble\fi
741   }@gobble
742 }
```

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

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

```
743 </package>
744 <pdftex-def>\MT@requires@pdftex5{
745 <*pdftex-def|luatex-def|xetex-def>
746   \def\MT@ledmac@setup{%
747     \ifMT@protrusion
748     \MT@ifdefined@c@TF\l@dunhbox@line{%
```

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

```
749   \MT@info@nl{Patching (e)ledmac to enable character protrusion}%
750   \let\MT@led@unhbox@line\l@dunhbox@line
751   \renewcommand*\l@dunhbox@line}[1]{%
752     \ifhbox##1%
753     \kern\leftmarginkern##1%
754     \expandafter\MT@led@unhbox@line\expandafter##1\expandafter
755     \kern\rightmarginkern##1%
756     \fi
757   }%
758 }{%
759   \MT@warning@nl{%
760     Character protrusion in paragraphs with line\MessageBreak
761     numbering will only work if you update ledmac}%
762 }%
763 \fi
764 }
```

```

765 </pdftex-def|lualatex-def|xetex-def>
766 <pdftex-def>
767 }{
768   \def\MT@ledmac@setup{%
769     \ifMT@protrusion
770       \MT@warning@n1{%
771         The pdftex version you are using does not allow\MessageBreak
772         character protrusion in paragraphs with line\MessageBreak
773         numbering by the 'ledmac' package.\MessageBreak
774         Upgrade pdftex to version 1.30 or later}%
775     \fi
776   }
777 }
778 </pdftex-def>

```

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

```

779 <*package|letterspace>
780 <*package>
781 \def\MT@restore@p@h{\chardef\%`%\% \chardef\#`#\# }

```

`\ifMT@unicode` Two new conditionals for use with `XYTeX` or `LuaTeX`.

```

\ifMT@fontspec 782 \newif\ifMT@unicode
783 \newif\ifMT@fontspec
784 \MT@with@package@T{xunicode}\MT@unicodetrue
785 \MT@with@package@T{fontspec}\MT@fontspectrue

```

`\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`.)

```

786 \def\MT@setupfont@hook{%

```

When a font is defined via `\fontspec`, the font is not actually loaded, hence `XYTeX` resp. `LuaTeX` would see a wrong font (in `\MT@get@slot`). Therefore, we load the current font.

```

787   \ifMT@fontspec\MT@font\fi

```

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

```

788   \MT@if@false
789   \MT@with@babel@and@T{spanish} \MT@if@true
790   \MT@with@babel@and@T{galician} \MT@if@true
791   \MT@with@babel@and@T{mexican} \MT@if@true
792   \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.)

```

793   \MT@with@package@T{csquotes}{%
794     \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`.

```

795   \MT@if@false
796   \MT@with@package@T{hyperref} \MT@if@true
797   \MT@with@package@T{tex4ht} \MT@if@true

```

```

798 \MT@with@package@T{mathastext}\MT@if@true
799 \ifMT@if@MT@restore@p@h\fi
800 }

```

Check again at the end of the preamble.

```

801 </package>
802 \MT@addto@setup{%
803 <*package>

```

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

```

804 \MT@with@package@T{pdfcprot}{%
805 \MT@error{Detected the `pdfcprot' package!\MessageBreak
806 \MT@MT' and `pdfcprot' may not be used together}{%
807 The `pdfcprot' package provides an interface to character protrusion.\MessageBreak
808 So does the `MT@MT' package. Using both packages at the same\MessageBreak
809 time will almost certainly lead to undesired results. Have your choice!}%
810 }%
811 \MT@with@package@T{ledmac}\MT@ledmac@setup
812 \MT@with@package@T{eledmac}\MT@ledmac@setup
813 \MT@with@package@T{xunicode}\MT@xunicodetrue
814 \MT@with@package@T{fontspec}\MT@fontspec@true

```

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

```

815 \let\MT@setupfont@hook\@empty
816 \ifMT@fontspec
817 \g@addto@macro\MT@setupfont@hook{\MT@font}%
818 \fi
819 \MT@if@false
820 \MT@with@babel@and@T{spanish} \MT@if@true
821 \MT@with@babel@and@T{galician}\MT@if@true
822 \MT@with@babel@and@T{mexican} \MT@if@true
823 \ifMT@if@
824 \g@addto@macro\MT@setupfont@hook{%
825 \MT@if@defined@c@T\percentsign{\let%\percentsign}}%
826 \fi
827 \MT@with@package@T{csquotes}{%
828 \ifpackage@later{csquotes}{2005/05/11}{%
829 \g@addto@macro\MT@setupfont@hook\@disablequotes
830 }{%
831 \MT@warning@n1{%
832 Should you receive warnings about unknown slot\MessageBreak
833 numbers, try upgrading the `csquotes' package}%
834 }%
835 }%

```

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.

```

836 \MT@if@false
837 </package>
838 <plain> \MT@requires@latex2{
839 \MT@with@package@T{hyperref}{%
840 \pdfstringdefDisableCommands{%
841 <*package>
842 \let\pickup@font\MT@orig@pickup@font
843 \let\text@microtype@context\@secondoftwo
844 \let\microtype@context\@gobble
845 </package>
846 \def\lststyle{\pdfstringdefWarn\lststyle}%
847 \def\textls#1{\pdfstringdefWarn\textls}%
848 }%
849 <package> \MT@if@true
850 }%
851 <plain> }\relax
852 <*package>

```

```

853 \MT@with@package@T{tex4ht}\MT@if@true
854 \MT@with@package@T{mathastext}\MT@if@true
855 \ifMT@if@lg@addto@macro\MT@setupfont@hook\MT@restore@p@h\fi

```

The listings package makes numbers and letters active,

```

856 \MT@with@package@T{listings}{%
857   \g@addto@macro\MT@cfg@catcodes{%
858     \MT@while@num{"30"}{"3A"}{\catcode\@tempcnta 12\relax}%
859     \MT@while@num{"41"}{"5B"}{\catcode\@tempcnta 11\relax}%
860     \MT@while@num{"61"}{"7B"}{\catcode\@tempcnta 11\relax}%
861   }%

```

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

```

862   \g@addto@macro\MT@setupfont@hook{%
863     \catcode`\z@

```

Inside a listing, \space is redefined.

```

864   \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.

```

865     \let\lst@ProcessLetter\@empty
866   }%
867 }%

```

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.

```

868 </package>
869 <plain> \MT@requires@latex2{
870   \MT@with@package@T{soul}{%
871     \soulregister\lststyle 0%
872     \soulregister\textls 1%
873   }%

```

Under plain T_EX, soul doesn't register itself the L^AT_EX way, hence we have to use a different test in this case.

```

874 <*plain>
875   }{\ifx\SOU@L@\undefined\else
876     \soulregister\lststyle 0%
877     \soulregister\textls 1%
878   \fi}%
879 </plain>
880 <*package>

```

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

```

881 \MT@with@package@T{pinyin}{%
882   \let\MT@orig@py@macron\py@macron
883   \ifpackage@later{pinyin}{2005/08/11}{% 4.6.0
884     \def\py@macron#1#2{%
885       \let\pickup@font\MT@orig@pickupfont
886       \MT@orig@py@macron{#1}{#2}%
887       \let\pickup@font\MT@pickupfont}%
888     }%
889     \def\py@macron#1{%
890       \let\pickup@font\MT@orig@pickupfont
891       \MT@orig@py@macron{#1}%
892       \let\pickup@font\MT@pickupfont}%
893   }%
894 }%

```

```

895 </package>
896 }
897 </package|letterspace>

```

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

```

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

```

14.2 Font setup

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

```

899 <*pdfTeX-def|xetex-def|luatex-def>
900 \def\MT@setupfont{\MT@setupfont@hook}

```

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

```

901 <pdfTeX-def>\MT@requires@pdfTeX7{
902 <pdfTeX-def|luatex-def>\g@addto@macro\MT@setupfont\MT@copy@font
903 <pdfTeX-def>}\relax

```

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

```

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

```

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

```

906   \MT@exp@one@n\MT@find@file\MT@family
907   \ifx\MT@familyalias\@empty \else
908     \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.)

```

909 % \ifx\f@encoding\cf@encoding\else\@enc@update\fi
910 }

```

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

```

911 <pdfTeX-def>\MT@requires@pdfTeX6
912 <luatex-def>\MT@requires@luatex3
913 <pdfTeX-def|luatex-def> {\g@addto@macro\MT@setupfont\MT@tracking}\relax
914 \g@addto@macro\MT@setupfont{%
915   \MT@check@font
916   \ifMT@inlist@
917 <debug>\MT@show@pdfannot2%
918   \else
919     \MT@vinfo{Setting up font `~\MT@font'\on@line}%

```

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

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

```

920   \MT@protrusion
921 <pdfTeX-def|luatex-def> \MT@expansion
922 }

```

Interword spacing and kerning (pdfTeX 1.40).

```

923 <*pdf $\text{\TeX}$ -def>
924 \MT@requires@pdf $\text{\TeX}$ 6{
925 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
926 }\relax
927 </pdf $\text{\TeX}$ -def>

```

Disable ligatures (pdf \TeX 1.30).

```

928 <pdf $\text{\TeX}$ -def>\MT@requires@pdf $\text{\TeX}$ 5{
929 <pdf $\text{\TeX}$ -def| $\text{\LaTeX}$ -def>\g@addto@macro\MT@setupfont\MT@noligatures
930 <pdf $\text{\TeX}$ -def>}\relax
931 \g@addto@macro\MT@setupfont{%

```

Debugging.

```

932 <debug>\MT@show@pdfannot1%

```

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

```

933 \MT@register@font
934 \fi
935 }
936 </pdf $\text{\TeX}$ -def| $\text{\XeTeX}$ -def| $\text{\LaTeX}$ -def>

```

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

```

937 <*pdf $\text{\TeX}$ -def| $\text{\LaTeX}$ -def>
938 \let\MT@copy@font\relax
939 <pdf $\text{\TeX}$ -def>\MT@requires@pdf $\text{\TeX}$ 7{
940 \def\MT@copy@font{%

```

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

```

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

```

\MT@font@orig pdf \TeX doesn't allow copying a font that has already been copied and expanded/letterspaced. Hence, we have to get the original.

```

942 \expandafter\ifx\MT@font@copy\relax
943 \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%
944 \expandafter\ifx\MT@font@orig\relax
945 \MT@exp@two@c\MT@glet\MT@font@orig\font@name
946 \else
947 \MT@exp@two@c\let\font@name\MT@font@orig
948 \fi
949 \global\MT@exp@two@c\pdfcopyfont\MT@font@copy\font@name
950 <debug>\MT@dinfo1{creating new copy: \MT@font@copy}%

```

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

```

951 \MT@map@clistc\MT@active@features{%
952 \MT@exp@cs\ifx\MT@\@nameuse\{MT@abbr@##1}\relax\else
953 \def\@tempa{##1}%
954 \MT@exp@cs\MT@map@tlistc\{MT@##1@doc@contexts}\MT@rem@from@list
955 \fi
956 }%
957 \fi
958 \MT@exp@two@c\let\MT@font\MT@font@copy

```

We only need the font identifier for letterspacing.

```

959 \let\font@name\MT@font@copy

```

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

```

960 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
961 }

```

`\MT@rem@from@list`

```

962 \def\MT@rem@from@list#1{%
963   \MT@exp@cs\ifx{MT@\@tempa @#1font@list}\relax\else
964     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
965       \MT@font \csname MT@\@tempa @#1font@list\endcsname
966   \fi
967 }
968 \pdfTeX-def\relax
969 \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`

`\MT@encoding`

`\MT@family`

`\MT@series`

`\MT@shape`

`\MT@size`

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

```

970 \*package
971 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
972   \def\MT@encoding{#1}%
973   \ifMT@fontspec
974     \edef\MT@family{\MT@scrubfeature#2()}\relax}%
975   \else
976     \def\MT@family{#2}%
977   \fi
978   \def\MT@series  {#3}%
979   \def\MT@shape   {#4}%
980   \def\MT@size    {#5}%

```

`\MT@familyalias`

Alias family?

```

981 \MT@ifdefined@n@TF{MT@\MT@family @alias}%
982   {\MT@let@cn\MT@familyalias{MT@\MT@family @alias}}%
983   {\let\MT@familyalias\@empty}%
984 }

```

`\MT@scrubfeature`

Remove one resp. all feature counters (`fontspec`).

`\MT@scrubfeatures`

```

985 \def\MT@scrubfeature#1(#2)#3\relax{#1}
986 \def\MT@scrubfeatures#1(#2)#3\relax{%
987   #1%
988   \ifx\relax#3\relax\else

```



```

989     \MT@scrubfeatures#3\relax
990     \fi
991 }

\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 992 \newif\ifMT@do
993 \def\MT@maybe@do#1{%
    (but only if the feature isn't globally set to false)
994     \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).

995     \MT@dotrue
996     \edef\@tempa{\csname MT@#1@setname\endcsname}%
997     \MT@map@clist@n{font,encoding,family,series,shape,size}{%
998         \MT@ifdefined@n@TF{MT@checklist@#1}%
999         {\csname MT@checklist@#1\endcsname}%
1000         {\MT@checklist@{#1}}}%
1001     {#1}%
1002     }%
1003     \else
1004     \MT@dofalse
1005     \fi
1006     \ifMT@do

\MT@feat stores the current feature.

1007     \def\MT@feat{#1}%
1008     \csname MT@set@#1@codes\endcsname
1009     \else
1010     \MT@vinfo{... No \@nameuse{MT@abbr@#1}}%
1011     \fi
1012 }

\MT@dinfo@list

1013 <debug>\def\MT@dinfo@list#1#2#3{\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: #2
1014 <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).

1015 \def\MT@checklist@#1#2{%
1016 <!debug> \MT@ifdefined@n@T
1017 <debug> \MT@ifdefined@n@TF
1018     {MT@#2list@#1@\@tempa}{%

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

1019     \expandafter\MT@exp@one@n\expandafter\MT@in@clist
1020     \csname MT@#1\expandafter\endcsname
1021     \csname MT@#2list@#1@\@tempa\endcsname
1022     \ifMT@inlist@
1023 <debug>\MT@dinfo@list{#2}{#1}{in}%
1024     \MT@dotrue
1025     \else
1026 <debug>\MT@dinfo@list{#2}{#1}{not in}%
1027     \MT@dofalse
1028     \expandafter\MT@clist@break
1029     \fi
1030     }%

```

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.

```
1031 <debug> {\MT@info@list{#2}{#1}{}}%
1032 }
```

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

```
1033 \def\MT@checklist@family#1{%
1034 <!debug> \MT@ifdefined@n@T
1035 <debug> \MT@ifdefined@n@TF
1036   {MT@#1list@family@\@tempa}{%
1037     \MT@exp@two@n\MT@in@clist
1038     \MT@family{\csname MT@#1list@family@\@tempa\endcsname}%
1039     \ifMT@inlist@
1040 <debug>\MT@info@list{#1}{family}{in}%
1041     \MT@dotrue
1042   \else
1043 <debug>\MT@info@list{#1}{family}{not in}%
1044     \MT@dofalse
1045     \ifx\MT@familyalias\@empty \else
1046       \MT@exp@two@n\MT@in@clist
1047       \MT@familyalias{\csname MT@#1list@family@\@tempa\endcsname}%
1048       \ifMT@inlist@
1049 <debug> \MT@info@list{#1}{family alias}{in}%
1050       \MT@dotrue
1051 <debug>\else\MT@info@list{#1}{family alias}{not in}%
1052       \fi
1053     \fi
1054     \fi
1055     \ifMT@do \else
1056       \expandafter\MT@clist@break
1057     \fi
1058   }%
1059 <debug> {\MT@info@list{#1}{family}{}}%
1060 }
```

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

```
1061 \def\MT@checklist@size#1{%
1062 <!debug> \MT@ifdefined@n@T
1063 <debug> \MT@ifdefined@n@TF
1064   {MT@#1list@size@\@tempa}{%
1065     \MT@exp@cs\MT@in@rlist{MT@#1list@size@\@tempa}%
1066     \ifMT@inlist@
1067 <debug>\MT@info@list{#1}{size}{in}%
1068     \MT@dotrue
1069   \else
1070 <debug>\MT@info@list{#1}{size}{not in}%
1071     \MT@dofalse
1072     \expandafter\MT@clist@break
1073     \fi
1074   }%
1075 <debug> {\MT@info@list{#1}{size}{}}%
1076 }
```

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

```
1077 \def\MT@checklist@font#1{%
1078 <!debug> \MT@ifdefined@n@T
1079 <debug> \MT@ifdefined@n@TF
1080   {MT@#1list@font@\@tempa}{%
```

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

```
1081   \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1082   \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter
1083   \@tempb \csname MT@#1list@font@\@tempa\endcsname
1084   \ifMT@inlist@
1085 <debug>\MT@info@list{#1}{font}{in}%
1086   \expandafter\MT@clist@break
```

```

1087     \else
1088 <debug>\MT@info@list{#1}{font}{not in}%
1089     \MT@dofalse
1090     \fi
1091   }%
1092 <debug> {\MT@info@list{#1}{font}{}}%
1093 }

```

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`.

```

1094 \newif\ifMT@nofamily
1095 </package>

```

`\MT@protrusion` Set up for protrusion?

```

1096 <*pdfTeX-def|xetex-def|luatex-def>
1097 \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.

```

1098 \def\MT@set@pr@codes{%
1099   \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).

```

1100   \MT@if@list@exists{%
1101     \ifMT@nofamily
1102       \MT@ifdefined@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1103         \MT@info@n@l{Loading generic settings for font family\MessageBreak
1104           '\MT@family' (encoding: \MT@encoding).\MessageBreak
1105           For optimal results, create family-specific settings.\MessageBreak
1106           See the microtype manual for details}%
1107         \MT@glet@nc{\MT@encoding-\MT@family-settings}\@empty
1108       }%
1109     \fi
1110     \MT@get@font@dimen@six{%
1111       \MT@get@opt
1112       \MT@reset@pr@codes

```

Get the name of the inheritance list and parse it.

```

1113     \MT@get@inh@list

```

Set an input encoding?

```

1114     \MT@set@inputenc{c}%

```

Load additional lists?

```

1115     \MT@load@list\MT@pr@c@name
1116     \MT@set@listname

```

Load the main list.

```

1117     \MT@let@cn@tempc{MT@pr@c@\MT@pr@c@name}%
1118     \expandafter\MT@set@codes\@tempc,\relax,%
1119   }\MT@reset@pr@codes
1120 }

```

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

`\MT@dimen@six`

```

1121 \def\MT@get@font@dimen@six{%
1122   \ifnum\fontdimen6\MT@font=\z@
1123     \MT@warning@n@l{%

```

```

1124     Font \MT@font does not specify its \MessageBreak
1125     \@backslashchar fontdimen 6 (width of an `em')! Therefore, \MessageBreak
1126     \@nameuse{MT@abbr@MT@feat} will not work with this font}%
1127     \expandafter\@gobble
1128   \else
1129     \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1130     \expandafter\@firstofone
1131   \fi
1132 }

\MT@set@all@pr    Set all protrusion codes of the font.
1133 \def\MT@set@all@pr#1#2{%
1134   (debug)\MT@dinfo@n1{3}{-- lp/rp: setting all to #1/#2}%
1135   \let\MT@temp@empty
1136   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
1137   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
1138   \MT@do@font\MT@temp
1139 }

\MT@reset@pr@codes@    All protrusion codes are zero for new fonts. However, if we have to reload the font
\MT@reset@pr@codes    due to different contexts, we have to reset them. This command will be changed by
                        \microtypecontext if necessary.
1140 \def\MT@reset@pr@codes@{\MT@set@all@pr\z@z@}
1141 \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.
1142 \def\MT@the@pr@code{\@tempcntb}
1143 (pdftex-def|luatex-def)
1144 (pdftex-def)\MT@requirespdftex6
1145 (luatex-def)\MT@requiresluatex3
1146   {\def\MT@the@pr@code@tr{%
1147     \numexpr\@tempcntb+\MT@letterspace@/2\relax
1148   }
1149 } \relax
1150 (pdftex-def|luatex-def)

\MT@set@codes    Split up the values and set the codes.
1151 \def\MT@set@codes#1,{%
1152   \ifx\relax#1\@empty\else
1153     \MT@split@codes #1==\relax
1154     \expandafter\MT@set@codes
1155   \fi
1156 }

\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.
1157 \def\MT@split@codes#1=#2=#3\relax{%
1158   \def\@tempa{#1}%
1159   \ifx\@tempa\@empty \else
1160     \MT@get@slot
1161     (pdftex-def|luatex-def) \ifnum\MT@char > \m@ne
1162     (xetex-def) \ifx\MT@char\@empty \else
1163       \MT@get@char@unit
1164       \csname MT@\MT@feat @split@val\endcsname#2\relax
1165     \fi
1166   \fi
1167 }

\MT@pr@split@val
1168 \def\MT@pr@split@val#1,#2\relax{%
1169   \def\@tempb{#1}%

```

```

1170 \MT@ifempty\@tempb\relax{%
1171   \MT@scale@to@em
1172   \lrcode\MT@font\MT@char=\MT@the@pr@code
1173 (debug)\MT@dinfol{4}{;;; lp (\MT@char): \number\lrcode\MT@font\MT@char\space: [#1]}%
1174 }%
1175 \def\@tempb{#2}%
1176 \MT@ifempty\@tempb\relax{%
1177   \MT@scale@to@em
1178   \rprcode\MT@font\MT@char=\MT@the@pr@code
1179 (debug)\MT@dinfol{4}{;;; rp (\MT@char): \number\rprcode\MT@font\MT@char\space: [#2]}%
1180 }%

```

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

```

1181 \MT@ifdefined@c@T\MT@pr@inh@name{%
1182   \MT@ifdefined@nT\MT@inh@\MT@pr@inh@name @\MT@char @}{%
1183     \MT@exp@cs\MT@map@tlist@c
1184     {MT@inh@\MT@pr@inh@name @\MT@char @}%
1185     \MT@set@pr@heirs
1186   }%
1187 }%
1188 }

```

`\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 `\lrcode` resp. `\rprcode`, since this would disallow protrusion factors larger than the character width (since `\[lr]rcode`’s limit is 1000). Now, the maximum protrusion is 1 em of the font.

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

```

1189 (pdfTEX-def)\MT@requires@pdfTEX3{
1190 \def\MT@scale@to@em{%
1191   \@tempcntb=\MT@count\relax

```

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

```

1192   \MT@scale\@tempcntb \@tempb \MT@dimen@six
1193   \ifnum\@tempcntb=\z@ \else
1194     \MT@scale@factor
1195   \fi
1196 }

```

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

```

1197 \def\MT@get@charwd{%
1198 (*pdfTEX-def)
1199 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1200 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1201 ^^Q \MT@count=\wd\z@
1202 (/pdfTEX-def)
1203 (luatEX-def) \MT@count=\fontcharwd\MT@font\MT@char\relax

```

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

```

1204 (*xETEX-def)
1205   \ifnum\MT@char@<\z@
1206     \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%

```

```

1207 \MT@count=\wd\z@
1208 \else
1209 \MT@count=\fontcharwd\MT@font\MT@char@\relax
1210 \fi
1211 /xetex-def
1212 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1213 }

```

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`.

```

1214 (*pdfTeX-def)
1215 \MT@requires@pdftex6{
1216 \g@addto@macro\MT@get@charwd{%
1217 \MT@ifdefined@cT\MT@letterspace@
1218 {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1219 }
1220 }\relax
1221 }{

```

No adjustment with versions 0.14f and 0.14g.

```

1222 \def\MT@scale@to@em{%
1223 \MT@count=\@tempb\relax
1224 \ifnum\MT@count=\z@ \else
1225 \MT@scale@factor
1226 \fi
1227 }

```

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

```

1228 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1229 }
1230 /pdfTeX-def
1231 /pdfTeX-def|xetex-def|luatex-def

```

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

```

1232 (*package)
1233 \def\MT@get@font@dimen#1{%
1234 \ifnum\fontdimen#1\MT@font=\z@
1235 \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1236 \@backslashchar fontdimen #1 (it's zero)! \MessageBreak
1237 You should use a different `unit' for \MT@curr@list@name}%
1238 \else
1239 \MT@count=\fontdimen#1\MT@font
1240 \fi
1241 }

```

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

```

1242 \def\MT@info@missing@char{%
1243 \MT@info@n1{Character '\the\MT@toks'
1244 ^^X \iffontchar\MT@font\MT@char@
1245 has a width of 0pt
1246 ^^X \else is missing\fi
1247 ^^Q \MessageBreak (it's probably missing)
1248 \MessageBreak in font '\MT@font'. \MessageBreak
1249 Ignoring protrusion settings for this character}%
1250 }

```

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

```

1251 \def\MT@scale@factor{%
1252 \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1253 \expandafter\MT@scale\expandafter \@tempcntb
1254 \csname MT@\MT@feat @factor@\endcsname \@m
1255 \fi
1256 \ifnum\@tempcntb>\csname MT@\MT@feat @max@\endcsname\relax

```

```

1257 \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @max}%
1258 \else
1259 \ifnum\@tempcntb<\csname MT@\MT@feat @min\endcsname\relax
1260 \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @min}%
1261 \fi
1262 \fi
1263 }

```

`\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.

```

1264 \def\MT@warn@code@too@large#1{%
1265 \@tempcnta=#1\relax
1266 \ifnum\csname MT@\MT@feat @factor\endcsname=\@m \else
1267 \expandafter\MT@scale\expandafter\@tempcnta\expandafter
1268 \@m \csname MT@\MT@feat @factor\endcsname
1269 \fi
1270 \MT@scale\@tempcnta \MT@dimen@six \MT@count
1271 \MT@warning@n1{The \@nameuse{MT@abbr@\MT@feat} code \@temp\space
1272 is too large for character\MessageBreak
1273 `the\MT@toks' in \MT@curr@list@name.\MessageBreak
1274 Setting it to the maximum of \number\@tempcnta}%
1275 \@tempcntb=#1\relax
1276 }

```

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

```

1277 \def\MT@get@opt{%
1278 \MT@set@listname

```

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

```

\MT@sp@factor@ 1279 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}{%
\MT@kn@factor@ 1280 \MT@let@nn{MT@\MT@feat @factor@}
1281 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}%
1282 \MT@vinfo{... : Multiplying \@nameuse{MT@abbr@\MT@feat} codes by
1283 \number\csname MT@\MT@feat @factor\endcsname/1000}%
1284 }{%
1285 \MT@let@nn{MT@\MT@feat @factor@}{MT@\MT@feat @factor}%
1286 }%

```

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

```

\MT@sp@unit@
\MT@kn@unit@ 1287 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}{%
1288 \MT@let@nn{MT@\MT@feat @unit@}%
1289 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}%
1290 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\@empty
1291 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1292 relative to character widths}%
1293 \else
1294 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\m@ne
1295 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1296 relative to width of space}%
1297 \fi
1298 \fi
1299 }{%
1300 \MT@let@nn{MT@\MT@feat @unit@}{MT@\MT@feat @unit}%
1301 }%

```

`\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.

```

1302 \let\MT@get@char@unit\relax
1303 \let\MT@get@space@unit@gobble

```

```

1304 \MT@exp@cs\ifx{MT@MT@feat @unit@}\@empty
1305 \let\MT@get@char@unit\MT@get@charwd
1306 \else
1307 \MT@exp@cs\ifx{MT@MT@feat @unit@}\m@ne
1308 \let\MT@get@space@unit\MT@get@font@dimen
1309 \else
1310 \MT@exp@cs\MT@get@unit{MT@MT@feat @unit@}%
1311 \fi
1312 \fi

```

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

```

1313 \MT@ifdefined@n@T{MT@MT@feat @c@\csname MT@MT@feat @c@name\endcsname @preset}{%
1314 \csname MT@preset@MT@feat\endcsname
1315 \MT@let@nc{MT@reset@MT@feat @codes}\relax
1316 }%
1317 }

```

`\MT@get@unit` If unit contains an em or ex, we use the corresponding `\fontdimen` to obtain the
`\MT@get@unit@` 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.

```

1318 \def\MT@get@unit#1{%
1319 \expandafter\MT@get@unit@#1 e!\@nil
1320 \ifx\x\@empty\else\let#1\x\fi
1321 \@defaultunits\@tempdima#1 pt\relax\@nnil
1322 \ifdim\@tempdima=\z@
1323 \MT@warning@n1{%
1324 Cannot set \@nameuse{MT@abbr@MT@feat} factors relative to zero\MessageBreak
1325 width. Setting factors of list ~\@nameuse{MT@MT@feat @c@name}\MessageBreak
1326 relative to character widths instead}%
1327 \let#1\@empty
1328 \let\MT@get@char@unit\MT@get@charwd
1329 \else
1330 \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} factors relative
1331 to \the\@tempdima}%
1332 \MT@count=\@tempdima\relax
1333 \fi
1334 }
1335 \def\MT@get@unit@#1e#2#3\@nil{%
1336 \ifx\#3\\\let\x\@empty \else
1337 \if m#2%
1338 \edef\x{#1\fontdimen6\MT@font}%
1339 \else
1340 \if x#2%
1341 \edef\x{#1\fontdimen5\MT@font}%
1342 \fi
1343 \fi
1344 }
1345 }

```

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

```
1346 \def\MT@set@inputenc#1{%
```

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

```

1347 \def\MT@cat{#1}%
1348 \edef\@tempa{MT@MT@feat @#1@\csname MT@MT@feat @#1@name\endcsname @inputenc}%
1349 \MT@ifdefined@n@T\@tempa\MT@set@inputenc
1350 }

```

`\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.

```

1351 \MT@addto@setup{%
1352 \ifpackageloaded{inputenc}{%

```



```

1353 \ifpackageafter{inputenc}{2006/02/22}{%
1354 \def\MT@set@inputenc{%
1355 \MT@ifstreq\inputencodingname{\csname\@tempa\endcsname}\relax
1356 \MT@load@inputenc
1357 }%
1358 }{%
1359 \let\MT@set@inputenc\MT@load@inputenc
1360 }%
1361 }{%
1362 \def\MT@set@inputenc{%
1363 \MT@warning@n1{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1364 \MessageBreak package isn't loaded. Ignoring input encoding}%
1365 }%
1366 }%
1367 }

```

`\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.

```

1368 \def\MT@load@inputenc{%
1369 \MT@cfg@catcodes
1370 <debug>\MT@dinfo@n1{loading input encoding: \@nameuse{\@tempa}}%
1371 \inputencoding{\@nameuse{\@tempa}}%
1372 }
1373 </package>

```

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

```

1374 <*pdfTeX-def|xetex-def|luatex-def>
1375 \def\MT@set@pr@heirs#1{%
1376 \lcode\MT@font #1 =\lcode\MT@font\MT@char\relax
1377 \rcode\MT@font #1 =\rcode\MT@font\MT@char\relax
1378 <debug>\MT@dinfo@n2{-- heir of \MT@char: #1}%
1379 <debug>\MT@dinfo@n4{;;; lp/rp (#1): \number\lcode\MT@font\MT@char\space/%
1380 <debug> \number\rcode\MT@font\MT@char\space}%
1381 }

```

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

`\MT@preset@pr@`

```

1382 \def\MT@preset@pr{%
1383 \expandafter\expandafter\expandafter\MT@preset@pr@
1384 \csname MT@pr@c@\MT@pr@c@name @preset\endcsname\@nil
1385 }
1386 \def\MT@preset@pr@#1,#2\@nil{%
1387 \ifx\MT@pr@unit@\empty
1388 \MT@warn@preset@twidth{pr}%
1389 \let\MT@preset@aux\MT@preset@aux@factor
1390 \else
1391 \def\MT@preset@aux{\MT@preset@aux@space2}%
1392 \fi
1393 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
1394 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
1395 \MT@set@all@pr\@tempa\@tempb
1396 }

```

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

`\MT@preset@aux@factor`

```

1397 \def\MT@preset@aux@factor#1#2{%

```

`\MT@preset@aux@space`

```

1398 \@tempcntb=#1\relax
1399 \MT@scale@factor
1400 \edef#2{\number\@tempcntb}%
1401 }
1402 \def\MT@preset@aux@space#1#2#3{%
1403 \def\@tempb{#2}%
1404 \MT@get@space@unit#1%
1405 \MT@scale@to@em
1406 \edef#3{\number\@tempcntb}%
1407 }

```

`\MT@warn@preset@twidth`

```

1408 \def\MT@warn@preset@twidth#1{%
1409   \MT@warning@n1{%
1410     Cannot preset characters relative to their widths\MessageBreak
1411     for \@nameuse{MT@abbr#1} list ` \@nameuse{MT@#1@c@name}'. Presetting them%
1412     \MessageBreak relative to lem instead}%
1413   }
1414 \</pdfTeX-def|XeTeX-def|LaTeX-def>

```

14.2.2 Expansion

`\MT@expansion` Set up for expansion?

```

1415 \<pdfTeX-def|LaTeX-def>
1416 \def\MT@expansion{\MT@maybe@do{ex}}

```

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

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

```

1417 \def\MT@set@ex@codes@{%
1418   \MT@if@list@exists{%
1419     \MT@get@ex@opt
1420     \let\MT@get@char@unit\relax
1421     \MT@reset@ef@codes
1422     \MT@get@inh@list
1423     \MT@set@inputenc{c}%
1424     \MT@load@list\MT@ex@c@name
1425     \MT@set@listname
1426     \MT@let@cn\@tempc{MT@ex@c@\MT@ex@c@name}%
1427     \expandafter\MT@set@codes\@tempc,\relax,%
1428     \MT@expandfont
1429   }\relax
1430 }
1431 \</pdfTeX-def|LaTeX-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.

```

1432 \<package>\newif\ifMT@nonselected
1433 \<pdfTeX-def|LaTeX-def>
1434 \def\MT@set@ex@codes@n{%
1435   \MT@nonselectedtrue
1436   \MT@if@list@exists
1437   \MT@get@ex@opt
1438   {%
1439     \let\MT@stretch@ \MT@stretch
1440     \let\MT@shrink@ \MT@shrink
1441     \let\MT@step@ \MT@step
1442     \let\MT@auto@ \MT@auto
1443     \let\MT@ex@factor@ \MT@ex@factor
1444   }%
1445   \MT@reset@ef@codes
1446   \MT@expandfont
1447   \MT@nonselectedfalse
1448 }

```

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

```

1449 \let\MT@set@ex@codes\MT@set@ex@codes@n

```

```

\MT@expandfont    Expand the font.
1450 \def\MT@expandfont{%
1451   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
1452 }

\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).
1453 \def\MT@set@all@ex#1{%
1454 <debug>\MT@dinfo@n1{3}{-- ex: setting all to \number#1}%
1455   \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
1456 }
1457 \def\MT@reset@ef@codes@{\MT@set@all@ex\MT@ex@factor@}

\MT@reset@ef@codes    However, this is only necessary for versions prior to 1.20.
1458 <*pdfTeX-def>
1459 \MT@requires@pdfTeX4{
1460   \def\MT@reset@ef@codes{%
1461     \ifnum\MT@ex@factor@=\@m \else
1462       \MT@reset@ef@codes@
1463     \fi
1464   }
1465 }{
1466 </pdfTeX-def>
1467   \let\MT@reset@ef@codes@\MT@reset@ef@codes@
1468 <pdfTeX-def>}

\MT@ex@split@val    There's only one number per character.
1469 \def\MT@ex@split@val#1\relax{%
1470   \@tempcntb=#1\relax

    Take an optional factor into account.
1471   \ifnum\MT@ex@factor@=\@m \else
1472     \MT@scale@\@tempcntb \MT@ex@factor@ \@m
1473   \fi
1474   \ifnum\@tempcntb > \MT@ex@max
1475     \MT@warn@ex@too@large\MT@ex@max
1476   \else
1477     \ifnum\@tempcntb < \MT@ex@min
1478       \MT@warn@ex@too@large\MT@ex@min
1479     \fi
1480   \fi
1481   \efcode\MT@font\MT@char=\@tempcntb
1482 <debug>\MT@dinfo@n1{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

    Heirs, heirs, I love thy heirs.
1483   \MT@ifdefined@c@T\MT@ex@inh@name{%
1484     \MT@ifdefined@nT{\MT@inh@\MT@ex@inh@name @\MT@char @}{%
1485       \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
1486     }%
1487   }%
1488 }

\MT@warn@ex@too@large
1489 \def\MT@warn@ex@too@large#1{%
1490   \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for
1491     character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
1492     Setting it to the maximum of \number#1}%
1493   \@tempcntb=#1\relax
1494 }

\MT@get@ex@opt    Apply different values to this font?
\MT@ex@factor@ 1495 \def\MT@get@ex@opt{%
\MT@stretch@ 1496   \MT@set@list@name
\MT@shrink@
\MT@step@
\MT@auto@

```

```

1497 \MT@ifdefined@n@TF{MT@ex@cc@MT@ex@cc@name @factor}{%
1498 \MT@let@cn\MT@ex@factor@{MT@ex@cc@MT@ex@cc@name @factor}%
1499 \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor@/1000}%
1500 }{%
1501 \let\MT@ex@factor@\MT@ex@factor
1502 }%
1503 \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
1504 \MT@get@ex@opt@{shrink}{Setting shrink limit to \number\MT@shrink@}%
1505 \MT@get@ex@opt@{step}{Setting expansion step to \number\MT@step@}%
1506 \def@tempa{autoexpand}%
1507 \MT@get@ex@opt@{auto}{\ifx\@tempa\MT@auto@ En\else Dis\fi abling automatic expansion}%
1508 \MT@ifdefined@n@T{MT@ex@cc@MT@ex@cc@name @preset}{%
1509 \MT@preset@ex
1510 \let\MT@reset@ef@codes\relax
1511 }%
1512 }

```

\MT@get@ex@opt@

```

1513 \def\MT@get@ex@opt@#1#2{%
1514 \MT@ifdefined@n@TF{MT@ex@cc@MT@ex@cc@name @#1}{%
1515 \MT@let@nn{MT@#1@}{MT@ex@cc@MT@ex@cc@name @#1}%
1516 \MT@vinfo{... : #2}%
1517 }{%
1518 \MT@let@nn{MT@#1@}{MT@#1}%
1519 }%
1520 }

```

\MT@set@ex@heirs

```

1521 \def\MT@set@ex@heirs#1{%
1522 \efcode\MT@font#1=\efcode\MT@font\MT@char
1523 <debug>\MT@dinfol{2}{-- heir of \MT@char: #1}%
1524 <debug>\MT@dinfol{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
1525 }

```

\MT@preset@ex

```

1526 \def\MT@preset@ex{%
1527 \@tempcntb=\csname MT@ex@cc@MT@ex@cc@name @preset\endcsname\relax
1528 \MT@scale@factor
1529 \MT@set@all@ex@\@tempcntb
1530 }
1531 </pdfTeX-def|LaTeX-def>

```

14.2.3 Interword spacing (glue)

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

```

1532 <*pdfTeX-def>
1533 \MT@requires@pdfTeX6{
1534 \def\MT@spacing{\MT@maybe@do{sp}}

```

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

```

1535 \def\MT@set@sp@codes{%
1536 \MT@if@list@exists{%
1537 \MT@get@font@dimen@six{%
1538 \MT@get@opt
1539 \MT@reset@sp@codes
1540 \MT@get@inh@list
1541 \MT@set@inputenc{c}%
1542 \MT@load@list\MT@sp@cc@name
1543 \MT@set@listname
1544 \MT@let@cn\@tempc{MT@sp@cc@MT@sp@cc@name}%
1545 \expandafter\MT@set@codes\@tempc,\relax,}%
1546 }\MT@reset@sp@codes
1547 }

```

```

\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).
1548 \def\MT@sp@split@val#1,#2,#3\relax{%
1549   \def\@tempb{#1}%
1550   \MT@ifempty\@tempb\relax{%
1551     \MT@get@space@unit2%
1552     \MT@scale@to@em
1553     \knbscode\MT@font\MT@char=\@tempcntb
1554   <debug>\MT@info@n1{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
1555   }%
1556   \def\@tempb{#2}%
1557   \MT@ifempty\@tempb\relax{%
1558     \MT@get@space@unit3%
1559     \MT@scale@to@em
1560     \stbscode\MT@font\MT@char=\@tempcntb
1561   <debug>\MT@info@n1{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
1562   }%
1563   \def\@tempb{#3}%
1564   \MT@ifempty\@tempb\relax{%
1565     \MT@get@space@unit4%
1566     \MT@scale@to@em
1567     \shbscode\MT@font\MT@char=\@tempcntb
1568   <debug>\MT@info@n1{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
1569   }%
1570   \MT@ifdefined@c@T\MT@sp@inh@name{%
1571     \MT@ifdefined@nT\MT@inh@\MT@sp@inh@name @\MT@char @}{%
1572       \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@sp@inh@name @\MT@char @}\MT@set@sp@heirs
1573     }%
1574   }%
1575 }

\MT@set@sp@heirs

1576 \def\MT@set@sp@heirs#1{%
1577   \knbscode\MT@font#1=\knbscode\MT@font\MT@char
1578   \stbscode\MT@font#1=\stbscode\MT@font\MT@char
1579   \shbscode\MT@font#1=\shbscode\MT@font\MT@char
1580   <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
1581   <debug>\MT@info@n1{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
1582   <debug>          \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
1583 }

\MT@set@all@sp

\MT@reset@sp@codes 1584 \def\MT@set@all@sp#1#2#3{%
\MT@reset@sp@codes@ 1585   <debug>\MT@info@n1{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
1586   \let\MT@temp\@empty
1587   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
1588   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
1589   \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
1590   \MT@do@font\MT@temp
1591 }
1592 \def\MT@reset@sp@codes@\MT@set@all@sp\z@\z@\z@
1593 \let\MT@reset@sp@codes\relax

\MT@preset@sp

\MT@preset@sp@ 1594 \def\MT@preset@sp{%
1595   \expandafter\expandafter\expandafter\MT@preset@sp@
1596   \csname MT@sp@c@\MT@sp@c@name @preset\endcsname\@nil
1597 }
1598 \def\MT@preset@sp@#1,#2,#3\@nil{%
1599   \ifx\MT@sp@unit@\@empty
1600     \MT@warn@preset@twidth{sp}%
1601     \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
1602     \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
1603     \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%

```

```

1604 \else
1605 \MT@ifempty{#1}{\let\tempa\empty}{\MT@preset@aux@space2{#1}\@tempa}%
1606 \MT@ifempty{#2}{\let\tempc\empty}{\MT@preset@aux@space3{#2}\@tempc}%
1607 \MT@ifempty{#3}{\let\tempb\empty}{\MT@preset@aux@space4{#3}\@tempb}%
1608 \fi
1609 \MT@set@all@sp\@tempa\@tempc\@tempb
1610 }
1611 }\relax

```

14.2.4 Additional kerning

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

```

1612 \MT@requires@pdftex6{
1613 \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

1614 \def\MT@set@kn@codes{%
1615 \MT@if@list@exists{%
1616 \MT@get@font@dimen@six{%
1617 \MT@get@opt
1618 \MT@reset@kn@codes
1619 \MT@get@inh@list
1620 \MT@set@inputenc{c}%
1621 \MT@load@list\MT@kn@c@name
1622 \MT@set@listname
1623 \MT@let@cn\@tempc{MT@kn@c@\MT@kn@c@name}%
1624 \expandafter\MT@set@codes\@tempc,\relax,}%
1625 }\MT@reset@kn@codes
1626 }

```

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

```

1627 \def\MT@kn@split@val#1,#2\relax{%
1628 \def\@tempb{#1}%
1629 \MT@ifempty\@tempb\relax{%
1630 \MT@get@space@unit2%
1631 \MT@scale@to@em
1632 \knbcode\MT@font\MT@char=\@tempcntb
1633 <debug>\MT@dinfo@n1{4}{;;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
1634 }%
1635 \def\@tempb{#2}%
1636 \MT@ifempty\@tempb\relax{%
1637 \MT@get@space@unit2%
1638 \MT@scale@to@em
1639 \knacode\MT@font\MT@char=\@tempcntb
1640 <debug>\MT@dinfo@n1{4}{;;; knac (\MT@char): \number\knacode\MT@font\MT@char: [#2]}%
1641 }%
1642 \MT@ifdefined@c@T\MT@kn@inh@name{%
1643 \MT@ifdefined@nT{MT@inh@\MT@kn@inh@name @\MT@char @}{%
1644 \MT@exp@cs\MT@map@tlist@c{MT@inh@\MT@kn@inh@name @\MT@char @}\MT@set@kn@heirs
1645 }%
1646 }%
1647 }

```

`\MT@set@kn@heirs`

```

1648 \def\MT@set@kn@heirs#1{%
1649 \knbcode\MT@font#1=\knbcode\MT@font\MT@char
1650 \knacode\MT@font#1=\knacode\MT@font\MT@char
1651 <debug>\MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
1652 <debug>\MT@dinfo@n1{4}{;;; knbc (#1): \number\knbcode\MT@font\MT@char/%
1653 <debug> \number\knacode\MT@font\MT@char}%
1654 }

```

`\MT@set@all@kn`

`\MT@reset@kn@codes`

`\MT@reset@kn@codes@`

```

1655 \def\MT@set@all@kn#1#2{%
1656 <debug>\MT@edinfo@n1{3}{-- knac/knbc: setting all to #1/#2}%
1657 \let\MT@temp\@empty
1658 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbcode\MT@font\@tempcnta=#1\relax}}%
1659 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knaccode\MT@font\@tempcnta=#2\relax}}%
1660 \MT@do@font\MT@temp
1661 }
1662 \def\MT@reset@kn@codes@{\MT@set@all@kn\z@\z@}
1663 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn@ 1664 \def\MT@preset@kn{%
1665 \expandafter\expandafter\expandafter\MT@preset@kn@
1666 \csname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
1667 }
1668 \def\MT@preset@kn@#1,#2\@nil{%
1669 \ifx\MT@kn@unit@\@empty
1670 \MT@warn@preset@twidth{kn}%
1671 \let\MT@preset@aux\MT@preset@aux@factor
1672 \else
1673 \def\MT@preset@aux{\MT@preset@aux@space2}%
1674 \fi
1675 \MT@ifempty{#1}\let\@tempa\@empty{\MT@preset@aux{#1}\@tempa}%
1676 \MT@ifempty{#2}\let\@tempb\@empty{\MT@preset@aux{#2}\@tempb}%
1677 \MT@set@all@kn\@tempa\@tempb
1678 }
1679 }\relax
1680 </pdfTeX-def>

```

14.2.5 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```

1681 <*pdfTeX-def|luatex-def>
1682 <pdfTeX-def>\MT@requires@pdfTeX6
1683 <luatex-def>\MT@requires@luatex3
1684 {

```

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

\MT@tracking@

```

\MT@tr@font@list 1685 \let\MT@tr@font@list\@empty
1686 \def\MT@tracking@{%
1687 \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
1688 \ifMT@inlist\@else
1689 \MT@maybe@do{tr}%
1690 \ifMT@do\@else
1691 \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
1692 \fi
1693 \fi
1694 }
1695 </pdfTeX-def|luatex-def>
1696 <pdfTeX-def|luatex-def|letterspace>\let\MT@tracking
1697 <pdfTeX-def|luatex-def> \MT@tracking@
1698 <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.

```

1699 <*pdfTeX-def|luatex-def|letterspace>
1700 \def\MT@set@tr@codes{%
1701 <*pdfTeX-def|luatex-def>
1702 \MT@vinfo{Tracking font `'\MT@font'\on@line}%
1703 \MT@get@font@dimen@six{%
1704 \MT@if@list@exists
1705 \MT@get@tr@opt

```

```

1706 \relax
1707 </pdfTeX-def|luatex-def>
1708 \MT@ifdefined@c@TF\MT@letterspace@relax{\let\MT@letterspace@
1709 \ifnum\MT@letterspace@=\z@

```

Zero tracking requires special treatment.

```

1710 \MT@set@tr@zero
1711 \else
1712 <pdfTeX-def|luatex-def> \MT@vinfo{... Tracking by \number\MT@letterspace@}%

```

Letterspacing only works in PDF mode.

```

1713 \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.

```

1714 \xdef\MT@lsfont{\csname\expandafter\string\font@name
1715 \number\MT@letterspace@ls\endcsname}%
1716 \expandafter\ifx\MT@lsfont\relax
1717 <debug>\MT@dinfo@n1{1}{... new letterspacing instance}%

```

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

```

1718 \MT@get@ls@basefont
1719 \global\expandafter\letterspacefont\MT@lsfont\font@name\MT@letterspace@

```

Scale interword spacing (not configurable in letterspace).

```

1720 <*pdfTeX-def|luatex-def>
1721 \MT@ifdefined@c@TF\MT@tr@ispace
1722 {\let\@tempa\MT@tr@ispace}%
1723 {\edef\@tempa{\MT@letterspace@*,,}}%
1724 \MT@ifdefined@c@TF\MT@tr@ospace
1725 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
1726 {\edef\@tempa{\@tempa,,,}}%
1727 \expandafter\MT@tr@set@space\@tempa,%
1728 </pdfTeX-def|luatex-def>
1729 <*letterspace>
1730 % spacing = <letterspace amount>*,,}
1731 \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@letterspace@relax sp
1732 * \fontdimen2\MT@lsfont/1000relax
1733 </letterspace>

```

Adjust outer kerning (microtype only).

```

1734 <*pdfTeX-def|luatex-def>
1735 \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
1736 \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

1737 \MT@ifdefined@c@TF\MT@tr@ligatures\MT@tr@noligatures
1738 </pdfTeX-def|luatex-def>
1739 <*letterspace>
1740 % no ligatures = {f}
1741 \tagcode\MT@lsfont`f=\m@ne
1742 </letterspace>

```

Adjust protrusion values now, and maybe later (in `\MT@pr@split@val`) (not for Lua \TeX , though, where the letterspaced font inherits the protrusion values from the base font).

```

1743 <debug>\MT@dinfo@n1{2}{... compensating for tracking (\number\MT@letterspace@)}%
1744 \MT@do@font{\lcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2relax
1745 \rcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2relax}%

```



```
1746 <pdfTeX-def> \let\MT@the@pr@code\MT@the@pr@code@tr
1747 \fi
```

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

```
1748 \aftergroup\MT@set@lsfont
1749 <pdfTeX-def|LaTeX-def> \let\MT@font\MT@lsfont
1750 <LaTeX-def> \ifMT@fontspec\MT@font\fi
```

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

```
\MT@curr@ls 1751 \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@letterspace@}}%
1752 \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.

```
1753 <*pdfTeX-def|LaTeX-def>
1754 \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
1755 \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
1756 \MT@tr@outer@l
1757 </pdfTeX-def|LaTeX-def>
```

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

```
1758 \ifx\MT@ls@adjust\@empty
1759 <letterspace> % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
1760 \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\font@name/2000\relax
1761 \MT@ls@outer@k
```

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

```
1762 <*pdfTeX-def|LaTeX-def>
1763 \else
1764 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
1765 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1766 \ifdim\MT@outer@kern=z@\else \MT@ls@outer@k \fi
1767 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
1768 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1769 </pdfTeX-def|LaTeX-def>
1770 <*letterspace>
1771 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
1772 \MT@afteraftergroup{%
1773 \MT@set@curr@ok
1774 \noexpand\MT@ls@outer@k
1775 }%
1776 </letterspace>
1777 \fi
1778 <*pdfTeX-def|LaTeX-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).

```
1779 \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.

```
1780 \MT@afteraftergroup{%
1781 \MT@set@curr@os
1782 \MT@set@curr@ok
1783 \noexpand\MT@tr@outer@r
1784 }%
1785 </pdfTeX-def|LaTeX-def>
1786 \fi
```

Since protrusion values are inherited in LuaTeX, we switch off the setup for this

font.

```
1787 <luatex-def> \MT@protrusionfalse
1788 <pdfTEX-def|luatex-def> }%
1789 }
```

`\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.)

```
1790 \def\MT@afteraftergroup#1{%
1791   \MT@ifdefined@n@TF{MT@aftergroup@number\currentgrouplevel}\relax{%
1792   \MT@exp@cs\xdef{MT@aftergroup@number\currentgrouplevel}%
1793   { \MT@exp@cs\MT@gllet{MT@aftergroup@number\currentgrouplevel}\noexpand\undefined#1}%
1794   \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
1795   {MT@aftergroup@number\currentgrouplevel}%
1796   }%
1797 }
1798 </pdfTEX-def|luatex-def|letterspace>
```

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

```
1799 <*pdfTEX-def|luatex-def>
1800 \def\MT@get@tr@opt{%
1801   \MT@set@listname
1802   \MT@ifdefined@n@T{MT@tr@cc@MT@tr@cc@name}{%
1803     \MT@let@cn\MT@letterspace{MT@tr@cc@MT@tr@cc@name}%
```

`\MT@tr@unit@` Different unit?

```
1804   \MT@ifdefined@n@T{MT@tr@cc@MT@tr@cc@name @unit}{%
1805     \MT@let@cn\MT@tr@unit@{MT@tr@cc@MT@tr@cc@name @unit}%
1806     \ifdim\MT@tr@unit@=1em
1807       \let\MT@tr@unit@\undefined
1808     \else
1809       \MT@let@cn@tempb{MT@tr@cc@MT@tr@cc@name}%
1810       \MT@get@unit\MT@tr@unit@
1811       \let\MT@tr@factor@\@m
1812       \MT@scale@to@em
1813       \edef\MT@letterspace{\number\@tempcntb}%
1814       \fi
1815     }%
1816   }%
```

`\MT@tr@ispace` Adjust interword spacing.

```
\MT@tr@ospace 1817 \MT@get@tr@opt@{spacing} {ispace}%
1818 \MT@get@tr@opt@{outerspacing}{ospace}%
```

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

```
1819 \MT@get@tr@opt@{outerkerning}{okern}%
```

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

```
1820 \MT@get@tr@opt@{noligatures} {ligatures}%
1821 }
```

`\MT@get@tr@opt@`

```
1822 \def\MT@get@tr@opt@#1#2{%
1823   \MT@ifdefined@n@T{MT@tr@cc@MT@tr@cc@name @#1}%
1824   { \MT@let@nn{MT@tr@#2}{MT@tr@cc@MT@tr@cc@name @#1}}%
1825 }
1826 </pdfTEX-def|luatex-def>
```

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

```
1827 <*pdfTEX-def|luatex-def|letterspace>
1828 <plain>\MT@requires@latex2{
1829 \def\MT@set@lsfont{\MT@exp@two@cc\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.

```
1830 \DeclareRobustCommand\lsstyle{%
1831   \not@math@alphabet\lsstyle\textls
1832   \let\glb@currsizel@empty
1833   pdftex-defluatex-def \def\MT@feat|%
1834   \let\MT@tracking\MT@set@tr@codes
1835   \selectfont
1836 }

```

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

```
1837 *plain
1838 }{
1839 \def\MT@set@lsfont{\MT@lsfont}
1840 \def\lsstyle{%
1841   \begingroup
1842   \escapechar@m@ne
1843   \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
1844   \MT@set@tr@codes
1845   \endgroup
1846 }
1847 \let\textls\undefined
1848 \let\lslig\undefined
1849 }
1850 /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.

```
1851 \DeclareRobustCommand\lslig[1]{%
1852   {\MT@ifdefined@c@TF\MT@curr@ls{%
1853     \escapechar@m@ne
1854     \MT@get@ls@basefont
1855     \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax
1856     \kern\MT@outer@kern
1857     \font@name #1%
1858     \kern\MT@outer@kern
1859   }{#1}}%
1860 }
```

`\MT@ls@basefont` pdf \TeX 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.

```
1861 \def\MT@get@ls@basefont{%
1862   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1863   \expandafter\ifx\MT@ls@basefont\relax
1864     \MT@exp@two@c\MT@gl@et\MT@ls@basefont\font@name
1865   \else
1866   debug\MT@dinfo@n1{1}{... fixing base font}%
1867     \MT@exp@two@c\let\font@name\MT@ls@basefont
1868   \fi
1869 }
```

`\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
1870 \def\MT@set@ls@basefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}
1871 \def\MT@set@tr@zero{%
1872   debug\MT@dinfo@n1{1}{... zero tracking}%
1873   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1874   \expandafter\ifx\MT@ls@basefont\relax \else
```

```

1875 <debug>\MT@info@n1{1}{... fixing base font}%
1876 \aftergroup\MT@set@lsbasefont
1877 \fi
1878 }
1879 </pdfTeX-def|luatex-def|letterspace>

```

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

```

1880 <*pdfTeX-def|luatex-def>
1881 <pdfTeX-def>\MT@requires@pdfTeX7{
1882 \def\MT@tr@noligatures{%
1883 \ifx\MT@tr@ligatures\@empty
1884 \MT@noligatures@\MT@lsfont\@undefined
1885 \else
1886 \MT@noligatures@\MT@lsfont\MT@tr@ligatures
1887 \fi
1888 }
1889 <*pdfTeX-def>
1890 }{
1891 \def\MT@tr@noligatures{%
1892 \MT@warning@n1{%
1893 Disabling selected ligatures is only possible since\MessageBreak
1894 pdfTeX 1.40.4. Disabling all ligatures instead}%
1895 \MT@gl@t\MT@tr@noligatures\relax
1896 }
1897 }
1898 </pdfTeX-def>

```

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

```
1899 \newskip\MT@outer@space
```

\MT@tr@set@space Adjust interword spacing (\fontdimen 2–4) 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.

```

1900 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
1901 <debug>\MT@info@n12{... orig. space: \the\fontdimen2\MT@lsfont,
1902 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
1903 <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
1904 \let\MT@temp\@empty
1905 \MT@tr@set@space@{#1}{#4}{2}\@empty
1906 \MT@tr@set@space@{#2}{#5}{3}\@pplus
1907 \MT@tr@set@space@{#3}{#6}{4}\@minus
1908 \MT@gl@t@nc{\MT@outer@space\expandafter\string\font@name}\MT@temp
1909 <debug>\MT@info@n12{... inner space: \the\fontdimen2\MT@lsfont,
1910 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
1911 <debug>\MT@info@n12{... outer space: \MT@temp}%
1912 }

```

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

```

1913 \def\MT@tr@set@space@#1#2#3#4{%
1914 \MT@ifempty{#2}{%
1915 \MT@ifempty{#1}{%
1916 \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
1917 }{%
1918 \MT@tr@set@space@@{#1}{#3}{1000}%
1919 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
1920 \fontdimen#3\MT@lsfont=\@tempdima
1921 }%
1922 }{%
1923 \MT@tr@set@space@@{#2}{#3}{2000}%
1924 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
1925 \MT@ifempty{#1}\relax{%
1926 \MT@tr@set@space@@{#1}{#3}{1000}%
1927 \fontdimen#3\MT@lsfont=\@tempdima

```

```

1928 }%
1929 }%
1930 }

```

`\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.

```

1931 \def\MT@tr@set@space@#1#2#3{%
1932   \MT@test@ast#1*\nil{%
1933     \MT@ifdefined@ccTF\MT@tr@unit@
1934     {\edef\@tempb{#1}\MT@scale@to@em}
1935     {\@tempcntb=#1\relax}%
1936     \@tempdima=\dimexpr \dimexpr\@tempcntb sp*\MT@dimen@six/1000\relax
1937     -\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).

```

1938   \ifnum#2=\tw@
1939     \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
1940   \fi
1941   \@tempdima=\dimexpr \fontdimen#2\MT@lsfont+\@tempdima\relax
1942 }{%
1943   \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
1944   \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
1945 }%
1946 <debug>\MT@dinfo@n13{... : font dimen #2 (#1): \the\@tempdima}%
1947 }

```

`\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).

```

1948 \def\MT@tr@outer@1{%
1949   \ifhmode
1950     \ifdim\lastskip>5sp
1951       \edef\x{\the\lastskip minus 0pt}%
1952       \setbox\z@\hbox{\MT@outer@space=\x}%
1953       \ifdim\wd\z@>\z@
1954 <debug>\MT@dinfo2{[[[ adjusting pre space: \the\MT@outer@space}%
1955         \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

1956     \let\MT@ls@outer@k\relax
1957   \else

```

The `ragged2e` package sets `\spaceskip` without glue.

```

1958     \ifdim\lastskip=%
1959       \ifnum\spacefactor<2000
1960         \spaceskip
1961       \else
1962         \ifdim\xspaceskip=\z@
1963           \dimexpr\spaceskip+\fontdimen7\font@name\relax
1964         \else
1965           \xspaceskip
1966         \fi
1967       \fi
1968 <debug>\MT@dinfo2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
1969         \unskip \hskip\MT@outer@space\relax
1970         \let\MT@ls@outer@k\relax
1971       \fi
1972     \fi
1973   \fi
1974 \fi
1975 }

```

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

disturb `t i k z`.

```
1976 \MT@addto@setup{%
1977   \ifpackageloaded{tikz}
1978     {\def\MT@tr@outer@r{%
1979       \ifnum\tikz@expandcount>\z@ \else
1980         \expandafter\MT@tr@outer@r\fi}}
1981   {\let\MT@tr@outer@r\MT@tr@outer@r}}
```

`\MT@tr@outer@next` The following is borrowed from `soul`. I've added the cases for italic correction, since tracking may also be triggered by text commands (e.g., `\textsc`).

```
1982 \def\MT@tr@outer@r{%
1983   \futurelet\MT@tr@outer@next\MT@tr@outer@r@@
1984 }
```

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

```
1985 \def\MT@if@outer@next#1{%
1986   \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
1987 }
```

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

```
1988 \def\MT@tr@outer@r@@{%
1989   \def\MT@temp*{%
```

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

```
1990   \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).

```
1991     \ifnum\currentgrouptype=10 \else
1992       \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
1993 <debug>\MT@dinfo2{}}] adjusting post space (1): \the\MT@outer@space}%
1994       \fi}%
1995       \expandafter\ifcat\expandafter\noexpand\cename \MT@tr@outer@next\endcename\egroup
1996       \ifhmode\unkern\fi\egroup
1997       \MT@set@curr@ok \MT@set@curr@os
1998       \def\MT@temp*{\afterassignment\MT@tr@outer@r@\let\MT@temp*}%
1999     \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.

```
2000     \MT@if@outer@next\maybe@ic{%
2001       \MT@set@curr@ok \MT@set@curr@os
2002       \def\MT@temp*{\afterassignment\MT@tr@outer@icr\let\MT@temp*}%
2003     }{%
```

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).

```
2004     \MT@if@outer@next\check@icr{%
2005       \def\MT@temp*{\aftergroup\MT@tr@outer@r@\check@icr\let\MT@temp*}%
2006     }{%
2007     \MT@if@outer@next\@sptoken{%
2008       \def\MT@temp* {\ifhmode\hskip\MT@outer@space
2009 <debug>\MT@dinfo2{}}] adjusting post space (2): \the\MT@outer@space}%
2010       \fi}%
2011     }{%
2012     \MT@if@outer@next~{%
2013     \def\MT@temp*~{\nobreak\hskip\MT@outer@space
```

```

2014 <debug>\MT@dinfo2{[]] adjusting post space (3): \the\MT@outer@space}%
2015         }%
2016     }{%
2017         \MT@if@outer@next\ \relax{%
2018         \MT@if@outer@next\space\relax{%
2019         \MT@if@outer@next\@xobeysp\relax{%

```

xspace requires special treatment.

```

2020         \MT@if@outer@next\xspace{%
2021         \def\MT@temp*\xspace{\futurelet\@let@token\MT@xspace}%
2022     }{%

```

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

```

2023         \def\MT@temp*\ifdim\MT@outer@kern=\z@else\MT@ls@outer@k
2024 <debug>\MT@dinfo2{--- adjusting post kern: \the\MT@outer@kern}%
2025         \fi}%
2026         \MT@let@nc{\MT@tr@outer@next}\relax
2027     }}}}]]\fi
2028 \fi\fi
2029 \MT@temp*%
2030 }

```

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

```

\MT@tr@outer@icr@ 2031 \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@\MT@tr@outer@r@}
2032 \def\MT@tr@outer@icr@{%
2033     \let\@let@token= \MT@tr@outer@next
2034     \maybe@ic@
2035 }

```

\MT@xspace If the group is followed by \xspace, we first feed \xspace with the next token, then check whether it has inserted a space.

```

2036 \def\MT@xspace{\@xspace@firsttrue\xspace
2037     \ifdim\lastskip>5sp
2038         \unskip \hskip\MT@outer@space
2039     \else
2040         \ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k \fi
2041     \fi
2042 }

```

For older pdfTeX versions and LuaTeX, throw an error.

```

2043 }{
2044     \DeclareRobustCommand\lsstyle{%
2045         \MT@error{Letterspacing only works with \MT@engine tex version
2046 <pdfTeX-def>         1.40%
2047 <luatex-def>         0.62%
2048         \MessageBreak or newer}
2049         {Upgrade \MT@engine tex, or try the `soul' package instead.}%
2050         \MT@glet\lsstyle\relax
2051     }
2052 }

```

And for XeTeX, too.

```

2053 </pdfTeX-def|luatex-def>
2054 <*xetex-def>
2055 \DeclareRobustCommand\lsstyle{%
2056     \MT@error{Letterspacing currently doesn't work with xetex}
2057     {Run pdfTeX or luatex, or use the `soul' package instead.}%
2058     \MT@glet\lsstyle\relax
2059 }
2060 </xetex-def>

```

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

```

2061 <package|letterspace>
2062 \DeclareRobustCommand\textls{%
2063   \ifstar{\let\MT@ls@adjust@MT@ls@adjust@empty\MT@textls}%
2064   {\let\MT@ls@adjust@MT@ls@adjust@relax\MT@textls}%
2065 }

```

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

```

2066 \newcommand\MT@textls[2][{}]{%
2067   \ifmmode
2068     \nfss@text{\MT@ls@set@ls{#1}\lssstyle#2}%
2069   \else
2070     \hmode@bgroup
2071     \MT@ls@set@ls{#1}%
2072     \lssstyle #2%
2073     \expandafter
2074     \egroup
2075   \fi
2076 }

```

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

```

\MT@ls@adjust@empty
\MT@ls@adjust@relax
\MT@ls@set@ls
2077 \def\MT@ls@adjust@empty{\let\MT@ls@adjust@empty}
2078 \def\MT@ls@adjust@relax{\let\MT@ls@adjust@relax}
2079 \def\MT@ls@set@ls#1{%
2080   \MT@ifempty{#1}%
2081   {\let\MT@letterspace@ \@undefined}%
2082   {\KV@sp@def\MT@letterspace@{#1}%
2083    \edef\MT@letterspace@{\number\MT@letterspace@}%
2084    \MT@ls@too@large\MT@letterspace@}%
2085   \MT@ls@adjust@
2086 }

```

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

```

2087 \def\MT@ls@too@large#1{%
2088   \ifnum#1>\MT@tr@max
2089     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
2090     \let#1\MT@tr@max
2091   \else
2092     \ifnum#1<\MT@tr@min
2093       \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
2094       \let#1\MT@tr@min
2095     \fi
2096   \fi
2097 }

```

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

```

\MT@tr@set@okern@
2098 \newdimen\MT@outer@kern
2099 </package|letterspace>
2100 <pdfTeX-def|luatex-def>
2101 \def\MT@tr@set@okern#1,#2,{%
2102   \let\MT@temp@empty
2103   \MT@ifempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
2104   \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
2105   \MT@glet@nc\MT@outer@kern\expandafter\string\font@name\MT@temp
2106   <debug>\MT@dinfo@n12{... outer kerning: (#1,#2)
2107   <debug> = \@nameuse{\MT@outer@kern\expandafter\string\font@name}}%
2108 }

```

`\MT@tr@set@okern@`

```

2109 \def\MT@tr@set@okern@#1{%
2110   \MT@test@ast#1*\@nil%

```



```

2111 \MT@ifdefined@TF\MT@tr@unit@
2112   {\edef\@tempb{#1}\MT@scale@to@em}
2113   {\@tempcntb=#1\relax}%
2114   \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
2115 }{%
2116 \MT@ifempty\@tempa{\let\@tempa\@m}\relax
2117 \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
2118   * \fontdimen6\MT@lsfont/2000\relax
2119 }%
2120 \advance\@tempdima -\dimexpr \MT@letterspace@ sp
2121   * \fontdimen6\MT@lsfont/2000\relax
2122 \edef\MT@temp{\MT@temp{\the\@tempdima}}%
2123 }
2124 </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.

```

2125 <*pdfTeX-def|luatex-def|letterspace>
2126 \def\MT@ls@outer@k{%
2127   \ifhmode
2128     \ifdim\lastkern=-3sp \unkern
2129     \ifdim\lastkern=3sp \kern-3sp
2130       \expandafter\expandafter\expandafter\@gobble
2131     \else \unkern
2132       \expandafter\expandafter\expandafter\@firstofone
2133     \fi
2134   \else
2135     \expandafter\@firstofone
2136   \fi
2137   {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
2138 \fi
2139 }
2140 </pdfTeX-def|luatex-def|letterspace>

```

14.2.6 Disabling ligatures

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

```

2141 <*pdfTeX-def|luatex-def>
2142 <pdfTeX-def>\MT@requires@pdfTeX5{
2143 \def\MT@noligatures{%
2144   \MT@dotrue
2145   \let\@tempa\MT@n1@setname
2146   \MT@map@clist@n{font,encoding,family,series,shape,size}{%
2147     \MT@ifdefined@TF{MT@checklist@##1}%
2148     {\csname MT@checklist@##1\endcsname}%
2149     {\MT@checklist@{##1}}%
2150     {n1}}%
2151 }%
2152 \ifMT@do
2153   \MT@noligatures@\MT@font\MT@n1@ligatures
2154 \fi
2155 }

```

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

```

2156 \def\MT@noligatures@#1#2{%
2157   \MT@ifdefined@TF#2{%

```

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

```

2158   \MT@ifdefined@TF\tagcode{%

```

No 'inputenc' key.

```

2159     \let\MT@warn@maybe@inputenc\@empty

```

```

2160 \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
2161 \MT@map@clist@c#2{%
2162 \KV@sp@def\@tempa{##1}\MT@get@slot
2163 \ifnum\MT@char>\m@ne \tagcode#1\MT@char=\m@ne \fi}%
2164 \MT@vinfo{... Disabling ligatures for characters: #2}%
2165 }{%
2166 \pdfnoligatures#1%
2167 \MT@warning{Cannot disable selected ligatures (pdftex doesn't\MessageBreak
2168 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
2169 the font instead}%
2170 }%
2171 }{%
2172 \pdfnoligatures#1%
2173 \MT@vinfo{... Disabling ligatures}%
2174 }%
2175 }
2176 <pdftex-def>\relax
2177 </pdftex-def|luatex-def>

```

14.2.7 Loading the configuration

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

```

2178 <*package>
2179 \def\MT@load@list#1{%
2180 \edef\@tempa{#1}%
2181 \MT@let@cn\@tempb{MT\MT@feat @c\@tempa @load}%
2182 \MT@ifstreq\@tempa\@tempb{%
2183 \MT@error{\@nameuse{MT@abbr@MT@feat} list \@tempa' cannot load itself}}%
2184 }{%
2185 \ifx\@tempb\relax \else
2186 \MT@ifdefined@n@TF{MT@MT@feat @c\@tempb}{%
2187 \MT@vinfo{... : First loading \@nameuse{MT@abbr@MT@feat} list \@tempb'}%
2188 \begingroup
2189 \MT@load@list\@tempb
2190 \endgroup
2191 \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list
2192 \noexpand\MessageBreak\@tempb'}%
2193 \MT@let@cn\@tempc{MT\MT@feat @c\@tempb}%
2194 \expandafter\MT@set@codes\@tempc,\relax,%
2195 }{%
2196 \MT@error{\@nameuse{MT@abbr@MT@feat} list \@tempb' undefined.\MessageBreak
2197 Cannot load it from list \@tempa'}}%
2198 }%
2199 \fi
2200 }%
2201 }

```

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

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

```

2202 \let\MT@file@list\empty
2203 \def\MT@find@file#1{%

```

Check for existence of the file only once.

```

2204 \MT@in@clist{#1}\MT@file@list
2205 \ifMT@inlist@ \else

```

Don't forget that because reading the files takes place inside a group, all commands that may be used there have to be defined globally.

```

2206 \MT@begin@catcodes
2207 \let\MT@begin@catcodes\relax
2208 \let\MT@end@catcodes\relax
2209 \InputIfFileExists{mt-#1.cfg}{%
2210 \edef\MT@curr@file{mt-#1.cfg}%

```

```

2211     \MT@vinfo{... Loading configuration file \MT@curr@file}%
2212     \MT@xadd\MT@file@list{#1,}%
2213   }{%
2214     \MT@get@basefamily#1\@empty\@empty\@empty\@nil
2215     \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
2216     \ifMT@inlist@
2217       \MT@xadd\MT@file@list{#1,}%
2218     \else
2219       \InputIfFileExists{mt-\@tempa.cfg}{%
2220         \edef\MT@curr@file{mt-\@tempa.cfg}%
2221         \MT@vinfo{... Loading configuration file \MT@curr@file}%
2222         \MT@xadd\MT@file@list{\@tempa,#1,}%
2223       }{%
2224         \MT@vinfo{... No configuration file mt-#1.cfg}%
2225         \MT@xadd\MT@file@list{#1,}%
2226       }%
2227     \fi
2228   }%
2229 \endgroup
2230 \fi
2231 }

```

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

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

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

```

2232 \def\MT@cfg@catcodes{%
2233   \makeatletter
2234   \catcode`\^7%
2235   \catcode`\ 9%
2236   \catcode`\^^I9%
2237   \catcode`\^^M9%
2238   \catcode`\z@
2239   \catcode`\{\@ne
2240   \catcode`\}\@tw@
2241   \catcode`\#6%
2242   \catcode`\%14%
2243   \MT@map@tlist@n
2244   {\!"\$\&'\'(\)*+,\-\.\/\:\;<|=|>?@[[]\_-|\|~}%
2245   \@makeother
2246 }

```

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

```

2247 \def\MT@begin@catcodes{%
2248   \begingroup
2249   \MT@cfg@catcodes
2250 }

```

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

```
2251 \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.

```
2252 \def\MT@get@basefamily#1#2#3#4\@nil{%
```

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	•	-	•	-	•	-	•	-	•	-	•	-	•	-	•	-

```

2253 \ifx\@empty#4%
2254 \def\@tempa{#1#2#3}%
2255 \else
2256 \let\@tempa\@empty
2257 \edef\@tempb{#1#2#3#4}%
2258 \expandafter\MT@get@basefamily@\@tempb\@nil
2259 \fi
2260 }

```

`\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’.

```

2261 \def\MT@get@basefamily@#1#2\@nil{%
2262 \edef\@tempa{\@tempa#1}%
2263 \ifx\#2\@expandafter\@gobble\else\expandafter\@firstofone\fi
2264 {\MT@in@tlist{#2}\MT@variants
2265 \ifMT@in@tlist\else\MT@get@basefamily@#2\@nil\fi}%
2266 }

```

`\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@
2267 \def\MT@get@listname#1{%
2268 (debug)\MT@dinfo@n1{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
2269 \let\MT@listname\@undefined
2270 \def\@tempb{#1}%
2271 \MT@map@tlist@c\MT@try@order\MT@get@listname@
2272 }
2273 \def\MT@get@listname@#1{%
2274 \expandafter\MT@next@listname#1%
2275 \ifx\MT@listname\@undefined \else
2276 \expandafter\MT@tlist@break
2277 \fi
2278 }

```

`\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.

```

2279 \def\MT@try@order{%
2280 {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
2281 {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
2282 }

```

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

```

2283 \def\MT@next@listname#1#2#3#4{%
2284 \ifnum#1=\z@\MT@nofamilytrue\fi
2285 \edef\@tempa{\MT@encoding
2286 /\ifnum#1=\@ne \MT@family \fi
2287 /\ifnum#2=\@ne \MT@series \fi
2288 /\ifnum#3=\@ne \MT@shape \fi
2289 /\ifnum#4=\@ne *\fi
2290 \MT@context}%
2291 (debug)\MT@dinfo@n1{1}{trying \@tempa}%

```

```

2292 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
2293 \MT@next@listname@#4%
2294 }{%

```

Also try with an alias family.

```

2295 \ifnum#1=\@ne
2296 \ifx\MT@familyalias\@empty \else
2297 \edef\@tempa{\MT@encoding
2298 /\MT@familyalias
2299 /\ifnum#2=\@ne \MT@series\fi
2300 /\ifnum#3=\@ne \MT@shape\fi
2301 /\ifnum#4=\@ne *\fi
2302 \MT@context}%
2303 <debug>\MT@info@n1{1}{(alias) \@tempa}%
2304 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
2305 \MT@next@listname@#4%
2306 }%
2307 \fi
2308 \fi
2309 }%
2310 }

```

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

```

2311 \def\MT@next@listname@#1{%
2312 \ifnum#1=\@ne
2313 \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @sizes}%
2314 \ifMT@inlist@
2315 \let\MT@listname\MT@size@name
2316 \fi
2317 \else
2318 \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
2319 \fi
2320 }

```

\MT@if@list@exists

```

\MT@context 2321 \def\MT@if@list@exists{%
2322 \MT@let@cn\MT@context{MT@\MT@feat @context}%
2323 \MT@ifstreq{@}\MT@context{\let\MT@context\@empty}\relax
2324 \MT@get@listname{\MT@feat @c}%
2325 \MT@ifdefined@c@TF\MT@listname{%
2326 \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
2327 \ifMT@nonselected
2328 \MT@vinfo{... Applying non-selected expansion (list `'\MT@listname')}'%
2329 \else
2330 \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list `'\MT@listname')}'%
2331 \fi
2332 \@firstoftwo
2333 }{%

```

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

```

2334 \MT@let@cn{MT@\MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```

2335 \ifMT@nonselected
2336 \MT@vinfo{... Applying non-selected expansion (no list)}%
2337 \else

```

Tracking doesn't require a list, either.

```

2338 \MT@ifstreq\MT@feat{tr}\relax{%
2339 \MT@warning{I cannot find a \@nameuse{MT@abbr@\MT@feat} list
2340 for font\MessageBreak`'\MT@font'%
2341 \ifx\MT@context\@empty\else\space(context: `'\MT@context')}\fi.
2342 Switching off\MessageBreak\@nameuse{MT@abbr@\MT@feat} for this font}%
2343 }%

```

```

2344     \fi
2345     \@secondoftwo
2346   }%
2347 }

\MT@get@inh@list    The inheritance lists are global (no context).
\MT@context 2348 \def\MT@get@inh@list{%
2349   \let\MT@context\@empty

2350   \MT@get@listname{\MT@feat @inh}%
2351   \MT@ifdefined@c@TF\MT@listname{%
2352     \MT@edefn{MT@\MT@feat @inh@name}{\MT@listname}%
2353   }%
2354   debug\MT@dinfo@n1{1}{... Using \nameuse{MT@abbr@\MT@feat} inheritance list
2355   debug         ^\MT@listname'}%
2356   \MT@let@cn\@tempc{MT@\MT@feat @inh@\MT@listname}%

   If the list is \@empty, it has already been parsed.
2356     \ifx\@tempc\@empty \else
2357   debug\MT@dinfo@n1{1}{parsing inheritance list ...}%

   The group is only required in case an input encoding is given.
2358     \begingroup
2359     \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak^\MT@listname'}%
2360     \MT@set@inputenc{inh}%
2361     \expandafter\MT@inh@do\@tempc,\relax,%
2362     \MT@gl@et@nc{MT@\MT@feat @inh@\MT@listname}\@empty
2363     \endgroup
2364     \fi
2365   }%
2366   \MT@let@nc{MT@\MT@feat @inh@name}\@undefined
2367 }%
2368 }

```

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@ 2369 \def\MT@get@slot{%
2370   \escapechar~\
2371   \let\MT@char@\m@ne
2372   \MT@noresttrue

```

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

```

2373   \MT@toks=\expandafter{\@tempa}%

```

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

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

```

2374   \expandafter\MT@is@letter\@tempa\relax\relax
2375   \ifnum\MT@char@ < \z@

```

- It might be an active character, i.e., an 8-bit character defined by `inputenc`. If so, we will expand it here to its LICR form.

```

2376   \MT@exp@two@c\MT@is@active\string\@tempa\@nil

```

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

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

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

```
2377 \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
2378 \MT@is@symbol
```

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

```
2379 {\expandafter\MT@is@composite\@tempa\relax\relax}%
2380 \ifnum\MT@char@ < \z@
```

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

```
2381 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2382 \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
2383 \fi
2384 \fi

2385 \let\MT@char\MT@char@
2386 \MT@get@slot@
2387 \escapechar\m@ne
2388 }
2389 </package>
```

$\backslash MT@get@slot@$

```
2390 <*pdfTeX-def|luatex-def|xetex-def>
2391 \def\MT@get@slot@{%
```

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

```
2392 <xetex-def> \ifnum\XeTeXfonttype\MT@font=\z@
2393 \ifnum\MT@char > \m@ne
```

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

```
2394 <*luatex-def>
2395 \ifnum\MT@char=47\relax
2396 \ifMT@noest \else
2397 \@tempcnta=\MT@lua{
2398 local glyph = microtype.name_to_slot([[ \expandafter\@gobble\@tempa ]],true)
2399 if glyph then tex.write(glyph)
2400 else tex.write(-1)
2401 end
2402 }\relax
2403 \ifnum\@tempcnta<\z@
2404 \MT@warn@unknown
2405 \let\MT@char\m@ne
2406 \else
2407 \edef\MT@char{\the\@tempcnta}%
2408 <debug>\MT@dinfo@n1{3}> \the\MT@toks' is a glyph name (\the\@tempcnta)%
2409 \fi
2410 \fi
2411 \else
2412 </luatex-def>
```

If the user has specified something like $\backslash fi$, 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.

```
2413 \ifMT@noest \else
```

```

2414     \MT@warn@rest
2415 (pdftex-def|luatex-def) \let\MT@char\m@ne
2416 (xetex-def) \let\MT@char\@empty
2417     \fi
2418 (luatex-def) \fi
2419     \else
2420     \MT@warn@unknown
2421     \fi
2422 (*xetex-def)
2423     \else

```

There are more possibilities for Xe_{La}TeX: It may also be a glyph name (prefixed with ‘/’). We indicate this to \MT@get@charwd by reversing the sign of \MT@char@.

```

2424     \ifnum\MT@char=47\relax
2425     \ifMT@norest \edef\MT@char{U47}%
2426     \else
2427     \@tempcnta=\XeTeXglyphindex"\expandafter\@gobble\@tempa"\relax
2428     \ifnum\@tempcnta=\z@
2429     \MT@warn@unknown
2430     \let\MT@char\@empty
2431     \else
2432     \edef\MT@char{\@tempa\space}%
2433     \edef\MT@char@{-\the\@tempcnta}%
2434 (debug)\MT@dinfol{3}{> `-\the\MT@toks' is a glyph name (\the\@tempcnta)}%
2435     \fi
2436     \fi
2437     \else
2438     \ifnum\MT@char > \m@ne
2439     \ifMT@norest

```

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

```

2440     \@tempcnta=\XeTeXcharglyph\MT@char\relax
2441     \ifnum\@tempcnta=\z@
2442     \MT@info@missing@char
2443     \let\MT@char\@empty
2444     \else
2445 (debug)\MT@dinfol{3}{> (glyph number: \the\@tempcnta,
2446 (debug) glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%
2447     \edef\MT@char{U\MT@char}%
2448     \fi
2449     \else
2450     \MT@warn@rest
2451     \let\MT@char\@empty
2452     \fi
2453     \else
2454     \MT@warn@unknown
2455     \let\MT@char\@empty
2456     \fi
2457     \fi
2458     \fi
2459 (/xetex-def)
2460 }
2461 (/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.

```

2462 (*luafile)
2463 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
2464     local slot_of_name = luaotfload.aux.slot_of_name
2465     microtype.name_to_slot = function(name, unsafe)
2466         return slot_of_name(font.current(), name, unsafe)
2467     end

```



```

2468 else
2469 -- we dig into internal structure (should be avoided)
2470 local function name_to_slot(name, unsafe)
2471   if fonts then
2472     local unicodes
2473     if fonts.ids then --- legacy luaotfload
2474       local tfmdata = fonts.ids[font.current()]
2475       if not tfmdata then return end
2476       unicodes = tfmdata.shared.otfdata.luatex.unicodes
2477     else --- new location
2478       local tfmdata = fonts.hashes.identifiers[font.current()]
2479       if not tfmdata then return end
2480       unicodes = tfmdata.resources.unicodes
2481     end
2482     local unicode = unicodes[name]
2483     if unicode then --- does the 'or' branch actually exist?
2484       return type(unicode) == "number" and unicode or unicode[1]
2485     end
2486   end
2487 end
2488 microtype.name_to_slot = name_to_slot
2489 end
2490
2491 (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` 2492 *(*pdf~~tex-def~~|~~luatex-def~~|~~xetex-def~~)*
2493 *\def\MT@max@char*
2494 *(pdf~~tex-def~~) {127 }*
2495 *(~~luatex-def~~|~~xetex-def~~) {1114111 }*
2496 *\def\MT@max@slot*
2497 *(pdf~~tex-def~~) {255 }*
2498 *(~~luatex-def~~|~~xetex-def~~) {1114111 }*
2499 *(/pdf~~tex-def~~|~~luatex-def~~|~~xetex-def~~)*

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

```

2500 (*package)
2501 \newif\ifMT@norest
2502 \def\MT@is@letter#1#2\relax{%
2503   \ifcat a\noexpand#1\relax
2504     \edef\MT@char@{\number`#1}%
2505     \ifx\#2\%
2506 (debug)\MT@dinfol{3}{> `the\MT@toks' is a letter (\MT@char@)}%
2507   \else
2508     \MT@norestfalse
2509   \fi
2510   \else
2511     \ifcat !\noexpand#1\relax
2512       \edef\MT@char@{\number`#1}%
2513 (debug)\MT@dinfol{3}{> `the\MT@toks' is a character (\MT@char@)}%
2514       \ifx\#2\%
2515         \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
2516       \else
2517         \MT@norestfalse
2518         \expandafter\MT@is@number#1#2\relax\relax
2519       \fi
2520     \fi
2521   \fi
2522 }

```

`\MT@is@number` Numbers may be specified as a three-digit decimal number (029), as a hexadecimal number (prefixed with " : "1D) or as an octal number (prefixed with ' : '35). They must consist of at least three characters (including the prefix), that is, "F is not

permitted.

```

2523 \def\MT@is@number#1#2#3\relax{%
2524   \ifx\relax#3\relax \else
2525     \ifx\relax#2\relax \else
2526       \MT@noesttrue
2527       \if#1"\relax
2528         \def\x{\uppercase{\edef\MT@char@{\number#1#2#3}}\x
2529 (debug)\MT@edinfo@n1{3}{> ... a hexadecimal number: \MT@char@}%
2530       \else
2531         \if#1'\relax
2532           \def\MT@char@{\number#1#2#3}%
2533 (debug)\MT@edinfo@n1{3}{> ... an octal number: \MT@char@}%
2534         \else
2535           \MT@ifint{#1#2#3}{%
2536             \def\MT@char@{\number#1#2#3}%
2537 (debug)\MT@edinfo@n1{3}{> ... a decimal number: \MT@char@}%
2538           }\MT@noestfalse
2539         \fi
2540       \fi
2541       \ifnum\MT@char@ > \MT@max@slot
2542         \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
2543         \let\MT@char@\m@ne
2544       \fi
2545     \fi
2546   \fi
2547 }
```

`\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.

```

2548 \def\MT@is@active#1#2\@nil{%
2549   \ifnum\catcode`#1 = \active
2550     \begingroup
2551     \set@display@protect
2552     \let\IeC\@firstofone
2553     \let\@inpen@undefined@\MT@undefined@char
```

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

```

2554   \def\UTFviii@defined##1{\ifx ##1\relax
2555     \MT@undefined@char{utf8}\else\expandafter ##1\fi}%
```

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

```

2556   \MT@ifdefined@c@T\PrerenderUnicode
2557   {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%
2558   \edef\x{\endgroup
2559   \def\noexpand\@tempa{\@tempa}%
```

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

```

2560   \MT@toks={\the\MT@toks\space(= \@tempa)}%
2561   }%
2562   \x
2563   \fi
2564 }
```

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

```

2565 \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.

```
2566 \def\MT@is@symbol{%
2567   \expandafter\def\expandafter\MT@char\expandafter
2568     {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%
2569   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2570     \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
2571   \ifnum\MT@char@ < \z@
```

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

```
2572   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
2573   \fi
2574 }
```

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

```
\MT@charstring 2575 \begingroup
2576   \catcode`\=\z@
2577   /MT@map@tlist@n{/CHARLEX}/@makeoether
2578   /lowercase{%
2579     /def/x{/endgroup
2580       /def/MT@charstring{\CHAR"%
2581       /def/MT@is@char##1\CHAR"##2##3##4/relax{%
2582         /ifx/relax##4/relax
2583         /ifMT@xunicode
2584         /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
2585         /relax/relax/relax/relax/relax
2586         /fi
2587         /else
2588         /ifx/relax##1/relax
2589         /if##3\relax
2590         /edef/MT@char@{/number"##2}%
2591         /MT@ifstreq/MT@charstring{##3##4}/relax/MT@noestfalse
2592         /else
2593         /edef/MT@char@{/number"##2##3}%
2594         /MT@ifstreq/MT@charstring{##4}/relax/MT@noestfalse
2595         /fi
2596         <debug> /MT@dinfo@n1{3}{> ~/the/MT@toks' is a \char (/MT@char@)}%
2597         /fi
2598         /fi
2599         }%
```

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

```
\MT@strip@prefix 2600 /def/MT@charxstring{\CHAR "%
\MT@is@charx 2601 /def/MT@strip@prefix##1>##2/relax{##2}%
2602 /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
2603 /ifx/relax##1/relax
2604 /ifx/relax##6/relax/else
2605 /edef/MT@char@{/number"##2##3##4##5}%
2606 /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
2607 <debug> /MT@dinfo@n1{3}{> ~/the/MT@toks' is a xunicode \char (/MT@char@)}%
2608 /fi
2609 /fi
2610 }%
2611 }%
2612 }
2613 /x
```

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

```
2614 \def\MT@is@composite#1#2\relax{%
2615   \ifx\#2\\\else
```

Again, we construct a control sequence, this time of the form: $\langle encoding \rangle \langle accent \rangle \langle character \rangle$, e.g., $\langle T1 \rangle \langle - \rangle \langle a \rangle$, 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.

```
2616   \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
2617     \string\csname\MT@encoding\endcsname
2618     \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%
2619   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
```

Again, xunicode.

```
2620   \ifnum\MT@char@ < \z@
2621     \ifMT@xunicode
2622       \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
2623       \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
2624         \MT@char\MT@charxstring\relax\relax\relax\relax\relax
2625     \fi
2626   \fi
2627 \fi
2628 }
```

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

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

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

Some warning messages, for performance reasons separated here.

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

```
\MT@set@list@name 2629 \def\MT@set@list@name{%
2630   \edef\MT@curr@list@name{\@nameuse{MT@abbr@\MT@feat} list\noexpand\MessageBreak
2631     ~\@nameuse{MT@\MT@feat @c@name}'}%
2632 }
```

`\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.

```
2633 \def\MT@warn@ascii{%
2634   \MT@warning@n1{Character ~\the\MT@toks' (= \MT@char@)
2635     is outside of ASCII range.\MessageBreak
2636     You must load the ~inputenc' package before using\MessageBreak
2637     8-bit characters in \MT@curr@list@name}%
2638 }
```

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

```
2639 \def\MT@warn@number@too@large#1{%
```

```

2640 \MT@warning@nl{%
2641   Number #1 in encoding `\'MT@encoding' too large!\MessageBreak
2642   Ignoring it in \MT@curr@list@name}%
2643 }

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

2644 \def\MT@warn@rest{%
2645   \MT@warning@nl{%
2646     Unknown slot number of character\MessageBreak`\'the\MT@toks'%
2647     \MT@warn@maybe@inputenc\MessageBreak
2648     in font encoding `\'MT@encoding'.\MessageBreak
2649     Make sure it's a single character\MessageBreak
2650     (or a number) in \MT@curr@list@name}%
2651 }

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

2652 \def\MT@warn@unknown{%
2653   \MT@warning@nl{%
2654     Unknown slot number of character\MessageBreak`\'the\MT@toks'%
2655     \MT@warn@maybe@inputenc\MessageBreak
2656     in font encoding `\'MT@encoding' in \MT@curr@list@name}%
2657 }

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

2658 \def\MT@warn@maybe@inputenc{%
2659   \MT@ifdefined@n@T
2660   {MT@\'MT@feat @\'MT@cat @\'csname MT@\'MT@feat @\'MT@cat @name\endcsname @inputenc}%
2661   { (input encoding `\'@nameuse
2662     {MT@\'MT@feat @\'MT@cat @\'csname MT@\'MT@feat @\'MT@cat @name\endcsname @inputenc}')}%
2663 }

```

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

For my reference:

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

Up to version 1.3 of this package, we were using `\define@newfont` as the hook, which is only called for *new* fonts, and therefore seemed the natural choice. However, this meant that we had to take special care to catch all fonts: we additionally

had to set up the default font, the error font (if it wasn't the default font), we had to check for some packages that might have been loaded before `microtype` and were loading fonts, e.g., `jurabib`, `ledmac`, `pi font` (loaded by `hyperref`), `tifa`, 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 2664 \let\MT@font@list\@empty
2665 \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`.

```
2666 </package>
2667 <*package|letterspace>
2668 <plain>\MT@requires@latex2{
2669 \MT@addto@setup{%
```

`\MT@orig@pickupfont` `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 expansion. (Tracking doesn't really work for other reasons.) Like us, CJK redefines `\pickup@font`.

```
2670 \ifpackageloaded{CJK}{%
2671 \ifpackageafter{CJK}{2006/10/17}% 4.7.0
2672 {\def\MT@orig@pickupfont{\CJK@ifundefined\CJK@plane}}%
2673 {\def\MT@orig@pickupfont{\ifundefined{CJK@plane}}}%
2674 \g@addto@macro\MT@orig@pickupfont
2675 {\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).

```
2676 \ifpackageloaded{CJKutf8}%
2677 {\ifpackageafter{CJKutf8}{2008/05/22}% 4.8.0
2678 {\ifpdf\expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}%
2679 {\@firstoftwo}}%
2680 {\@firstoftwo}%
2681 {\g@addto@macro\MT@orig@pickupfont{%
2682 {\expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2683 \define@newfont\else\edef\font@name{%
2684 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
2685 {\g@addto@macro\MT@orig@pickupfont{%
2686 {\expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2687 \define@newfont\def\CJK@temp{v}%
2688 \ifx\CJK@temp\CJK@plane
2689 \expandafter\ifx\csname CJK@cmaph@f@family\CJK@plane\endcsname\relax
2690 \else\csname CJK@cmaph@f@family\CJK@plane\endcsname\fi
2691 \else \CJK@addcmap\CJK@plane \fi
2692 \else\edef\font@name{%
2693 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
2694 }%
2695 \def\MT@orig@pickupfont{\expandafter\ifx\font@name\relax\define@newfont\fi}%
2696 }%
```

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

```
2697 \ifx\pickup@font\MT@orig@pickupfont \else
2698 \MT@warning@n1{%
2699 Command \string\pickup@font\space is not defined as expected.%
```

```

2700     \MessageBreak Patching it anyway. Some things may break%
2701 <*package>
2702     .\MessageBreak Double-check whether micro-typography is indeed%
2703     \MessageBreak applied to the document.%
2704     \MessageBreak (Hint: Turn on `verbose' mode)%
2705 </package>
2706     }%
2707     \fi

```

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

```

2708     \g@addto@macro\pickup@font{\begingroup}%

```

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

```

2709     \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
2710     \g@addto@macro\pickup@font{%
2711         \escapechar\m@ne
2712 <*package>
2713 <debug>         \global\MT@inannottrue
2714 <debug>         \MT@glet\MT@pdf@annot\@empty
2715 <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.

```

2716     \MT@let@cn\MT@font{MT@subst@expandafter\string\font@name}%
2717     \ifx\MT@font\relax
2718         \let\MT@font\font@name
2719     \else
2720         \ifx\MT@font\font@name \else
2721 <debug>         \MT@addto@annot{= substituted with \MT@font}%
2722         \MT@register@subst@font
2723     \fi
2724     \fi
2725     \MT@setupfont
2726 </package>
2727 <letterspace>     \MT@tracking
2728     \endgroup
2729     }%
2730 <*package>

```

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

```

2731     \let\MT@pickupfont\pickup@font

```

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

```

2732     \g@addto@macro\do@subst@correction
2733     {\edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
2734     \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` would break the accent if different fonts are used for the base character and the accent. Fortunately, L^AT_EX takes care that the fonts used for the `\accent` are already set up, so that we cannot be overlooking them.

```

2735     \let\MT@orig@add@accent\add@accent
2736     \def\add@accent#1#2{%
2737         \let\pickup@font\MT@orig@pickupfont

```

```

2738 \MT@orig@add@accent{#1}{#2}%
2739 \let\pickup@font\MT@pickupfont
2740 }%
2741 </package>
2742 }
2743 <plain>\relax
2744 <*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.

```
2745 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}
```

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

```
2746 \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).

```
2747 \def\MT@register@subst@font{\MT@exp@one@n\MT@in@clist\font@name\MT@font@list
2748 \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.

```
2749 \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.

```

2750 \def\MT@check@font@cx{%
2751 \MT@if@true
2752 \MT@map@clist@c\MT@active@features{%
2753 \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
2754 \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2755 \ifMT@inlist@
2756 \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
2757 \else
2758 \MT@if@false
2759 \fi
2760 }%
2761 \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
2762 }

```

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

```

2763 \def\MT@register@subst@font@cx{%
2764 \MT@map@clist@c\MT@active@features{%
2765 \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
2766 \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2767 \ifMT@inlist@ \else
2768 \MT@exp@cs\MT@xadd
2769 {MT@##1@\csname MT@##1@context\endcsname font@list}%
2770 {\font@name,}%
2771 \fi
2772 }%
2773 }

```

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

```

2774 \def\MT@register@font@cx{%
2775 \MT@map@clist@c\MT@active@features{%
2776 \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else

```



```

2777 \MT@exp@cs\MT@xadd
2778 {MT@##1@\csname MT@##1@context\endcsname font@list}%
2779 {\MT@font,}%
2780 \def\@tempa{##1}%
2781 \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
2782 \fi
2783 }%
2784 }

```

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

```

2785 \def\MT@maybe@rem@from@list#1{%
2786 \MT@ifstreq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
2787 \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
2788 \MT@font \csname MT@\@tempa @#1font@list\endcsname
2789 }%
2790 }

```

`\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.

```

2791 \def\microtypecontext#1{\MT@addto@setup{\microtypecontext{#1}}
2792 \MT@addto@setup{%
2793 \DeclareRobustCommand\microtypecontext[1]{%
2794 \MT@setup@contexts
2795 \let\MT@reset@context\relax

```

We need to ensure that math fonts are set up anew.

```

2796 \let\glb@currsiz@empty
2797 \setkeys{MTC}{#1}%
2798 \selectfont
2799 \MT@reset@context
2800 }%
2801 }

```

`\textmicrotypecontext` This is just a wrapper around `\microtypecontext`.

```

2802 \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@`

```

2803 \def\MT@reset@context@{%
2804 \MT@vinfo{<<< Resetting contexts\on@line
2805 <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
2806 <debug> / \MT@tr@context/\MT@kn@context/\MT@sp@context
2807 }%
2808 \selectfont
2809 }

```

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

```

2810 \def\MT@setup@contexts{%
2811 \MT@map@clist@c\MT@active@features
2812 {\MT@gl@et@c{MT@##1@font@list}\MT@font@list}%
2813 \MT@gl@et\MT@check@font\MT@check@font@cx
2814 \MT@gl@et\MT@register@font\MT@register@font@cx
2815 \MT@gl@et\MT@register@subst@font\MT@register@subst@font@cx
2816 \MT@gl@et\MT@setup@contexts\relax
2817 }

```

Define context keys.

```

2818 \MT@map@clist@c\MT@features@long{%
2819 \define@key{MTC}{#1}[]{}%
2820 \edef\@tempb{\@nameuse{MT@rbba#1}}%
2821 \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features

```

```
2822 \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).

```
2823 \MT@ifempty{##1}{\def\MT@val{0}}{\def\MT@val{##1}}%
2824 \MT@exp@cs\ifx{MT@\@tempb @context}\MT@val
2825 <debug>\MT@dinfo{1}{>>> no change of #1 context: ~\MT@val'}%
2826 \else
2827 \MT@vinfo{>>> Changing #1 context to ~\MT@val'\MessageBreak\n@nline
2828 <debug> \space(previous: ~\@nameuse{MT@\@tempb @context}')}%
2829 }%
2830 \def\MT@reset@context{\aftergroup\MT@reset@context@}%
```

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

```
2831 \MT@gl@et@nn{MT@reset@\@tempb @codes}{MT@reset@\@tempb @codes@}%
```

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

```
2832 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
2833 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
2834 \ifMT@inlist@ \else
2835 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{\MT@val}%
2836 <debug> \MT@dinfo{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
2837 \fi
2838 \MT@edef@n{MT@\@tempb @context}{\MT@val}%
2839 \fi
2840 \fi
2841 }%
2842 }
```

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

```
\MT@ex@context 2843 \MT@exp@one@n\MT@map@clist@{\MT@features,nl}{%}
```

```
\MT@tr@context 2844 \MT@def@n{MT@#1@context}{@}%
```

```
\MT@sp@context 2845 \MT@def@n{MT@#1@doc@contexts}{{@}}%
```

```
\MT@kn@context 2846 }
```

```
\MT@kn@context 2847 \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.

```
2848 \def\DeclareMicrotypeSet{%
2849 \ifstar
2850 \MT@DeclareSetAndUseIt
2851 \MT@DeclareSet
2852 }
```

```
\MT@DeclareSet
```

```
2853 \newcommand\MT@DeclareSet[3] [] {%
2854 \KV@sp@def\@tempa{#1}%
2855 \MT@ifempty\@tempa{%
2856 \MT@map@clist@c\MT@features{\MT@declare@sets{##1}{##2}{##3}}%
2857 }%
2858 \MT@map@clist@c\@tempa{%
2859 \KV@sp@def\@tempa{##1}%
2860 \MT@ifempty\@tempa\relax%
2861 \MT@is@feature{set declaration ~#2'}%
```

```

2862     \MT@exp@one@n\MT@declare@sets
2863     {\csname MT@rbba@\@tempa\endcsname}{#2}{#3}%
2864     }%
2865     }%
2866     }}%
2867     }%
2868 }

```

\MT@DeclareSetAndUseIt

```

2869 \newcommand\MT@DeclareSetAndUseIt[3] [] {%
2870   \MT@DeclareSet[#1]{#2}{#3}%
2871   \UseMicrotypeSet[#1]{#2}%
2872 }

```

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

```
2873 \let\MT@curr@set@name\empty
```

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

```

2874 \def\MT@declare@sets#1#2#3{%
2875   \KV@sp@def\MT@curr@set@name{#2}%
2876   \MT@ifdefined@n@T{MT@#1@set@\MT@curr@set@name}{%
2877     \MT@warning{Redefining \@nameuse{MT@abbr@#1} set ~\MT@curr@set@name'}%
2878     \MT@gl@et@nc{MT@#1@list@size@\MT@curr@set@name}\@empty
2879   }%
2880   \MT@gl@et@nc{MT@#1@set@\MT@curr@set@name}\@empty
2881   <debug>\MT@din@fo{1}{declaring \@nameuse{MT@abbr@#1} set ~\MT@curr@set@name'}%
2882   \setkeys{MT@#1@set}{#3}%
2883 }

```

\MT@define@set@key@ <#1> = font axis, <#2> = feature.

```

2884 \def\MT@define@set@key@#1#2{%
2885   \define@key{MT@#2@set}{#1} [] {%
2886     \MT@gl@et@nc{MT@#2@list@#1@\MT@curr@set@name}\@empty
2887     \MT@map@clist@n{##1}{%
2888       \KV@sp@def\MT@val{###1}%
2889       \MT@get@highlevel{#1}%

```

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

```

2890     \MT@exp@two@n@g@addto@macro
2891     {\csname MT@#2@list@#1@\MT@curr@set@name\expandafter\endcsname}%
2892     {\MT@val,}%
2893   }%

```

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

```

2894   \expandafter\g@addto@macro\expandafter\MT@font@sets
2895   \csname MT@#2@list@#1@\MT@curr@set@name\endcsname
2896   <debug>\MT@din@fo@n1{1}{-- #1: \@nameuse{MT@#2@list@#1@\MT@curr@set@name}}%
2897   }%
2898 }

```

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

```

2899 \def\MT@get@highlevel#1{%
2900   \expandafter\MT@test@ast\MT@val*\@nil\relax%

```

And ‘family = *’ will become \familydefault.

```

2901   \MT@ifempty\@tempa{\def\@tempa{#1}}\relax
2902   \edef\MT@val{\expandafter\noexpand\csname \@tempa default\endcsname}%

```

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

```

2903   }%
2904 }

```

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

```
2905 \def\MT@test@ast#1*#2\@nil{%
2906   \def\@tempa{#1}%
2907   \MT@ifempty{#2}%
2908 }
```

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

```
2909 \let\MT@font@sets\@empty
2910 \def\MT@fix@font@set#1{%
2911   \xdef#1{#1}%
2912   \ifMT@fontspec
2913     \xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%
2914   \fi
2915   \global\@onelevel@sanitize#1%
2916 }
```

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

```
2917 \def\MT@define@set@key@size#1{%
2918   \define@key{MT@#1@set}{size}[]{}%
2919   \MT@map@clist@n{##1}{%
2920     \KV@sp@def\MT@val{###1}%
2921     \expandafter\MT@get@range\MT@val--\@nil
2922     \ifx\MT@val\relax \else
2923       \MT@exp@cs\MT@xadd
2924         {MT@#1list@size@MT@curr@set@name}%
2925         {{{\MT@lower}{\MT@upper}\relax}}%
2926     \fi
2927   }%
2928   (debug)\MT@dinfo@n1{1}{-- size: \@nameuse{MT@#1list@size@MT@curr@set@name}}%
2929   }%
2930 }
```

Font sizes may also be specified as ranges. This has been requested by Andreas Böhmann, who has also offered valuable help in implementing this. Now, it is for instance possible to set up different lists for fonts with optical sizes. (The MinionPro project is trying to do this for the OpenType version of Adobe's Minion. See <http://developer.berlios.de/projects/minionpro/>.)

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

```
\MT@lower 2931 \def\MT@get@range#1-#2-#3\@nil{%
2932   \MT@ifempty{#1}{%
2933     \MT@ifempty{#2}{%
2934       \let\MT@val\relax
2935     }%
2936     \def\MT@lower{0}%
2937     \def\MT@val{#2}%
2938     \MT@get@size
2939     \edef\MT@upper{\MT@val}%
2940   }%
2941 }%
2942 \def\MT@val{#1}%
2943 \MT@get@size
2944 \ifx\MT@val\relax \else
2945   \edef\MT@lower{\MT@val}%
2946   \MT@ifempty{#2}{%
2947     \MT@ifempty{#3}%
2948     {\def\MT@upper{-1}}%

```

2048 pt is \TeX 's maximum font size.

```
2949   {\def\MT@upper{2048}}%
```

```

2950 }{%
2951   \def\MT@val{#2}%
2952   \MT@get@size
2953   \ifx\MT@val\relax \else
2954     \MT@ifdim\MT@lower>\MT@val{%
2955       \MT@error{%
2956         Invalid size range (\MT@lower\space > \MT@val) in font set
2957         ~\MT@curr@set@name'.\MessageBreak Swapping sizes}}{%
2958       \edef\MT@upper{\MT@lower}%
2959       \edef\MT@lower{\MT@val}%
2960     }{%
2961       \edef\MT@upper{\MT@val}%
2962     }%
2963     \MT@ifdim\MT@lower=\MT@upper
2964     {\def\MT@upper{-1}}%
2965     \relax
2966   \fi
2967 }%
2968 \fi
2969 }%
2970 }

```

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

```

2971 \def\MT@get@size{%
  A single star would mean \sizedefault, which doesn't exist, so we define it to be
  \normalsize.
2972   \if*\MT@val\relax
2973     \def\@tempa{\normalsize}%
2974   \else
2975     \MT@let@cn\@tempa{\MT@val}%
2976   \fi
2977   \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 `\setfontsize`, and not `\@setfontsize` because some classes might define the size selection commands by simply using `\fontsize` (e.g., the `a0poster` class).

```

2978   \begingroup
2979     \def\setfontsize##1##2##3##4\@nil{\endgroup\def\MT@val{##2}}%
2980     \@tempa\@nil
2981   \fi

```

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

```

2982   \MT@ifdimen\MT@val{%
2983     \@defaultunits\@tempdima\MT@val pt\relax\@nnil
2984     \edef\MT@val{\strip@pt\@tempdima}%
2985   }{%
2986     \MT@warning{Could not parse font size ~\MT@val'\MessageBreak
2987       in font set ~\MT@curr@set@name'}%
2988     \let\MT@val\relax
2989   }%
2990 }

```

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

```

2991 \def\MT@define@set@key@font#1{%
2992   \define@key{MT@#1@set}{font}[]{}%
2993   \MT@gl@et@nc{MT@#1@list@font@\MT@curr@set@name}\@empty
2994   \MT@map@cl@ist@n{##1}{%
2995     \KV@sp@def\MT@val{###1}%
2996     \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
2997     \expandafter\MT@get@font\MT@val///// \@nil

```

```

2998     \MT@exp@two@n@g@addto@macro
2999     {\csname MT@#1list@font@MT@curr@set@name\expandafter\endcsname}%
3000     {\MT@val,}%
3001     }%
3002     \expandafter@g@addto@macro\expandafter\MT@font@sets
3003     \csname MT@#1list@font@MT@curr@set@name\endcsname
3004 (debug)\MT@dinfol{1}{-- font: \@nameuse{MT@#1list@font@MT@curr@set@name}}%
3005     }%
3006 }

```

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

```

3007 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
3008   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
3009   \ifx\MT@val\relax\def\MT@val{0}\fi
3010   \expandafter@g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
3011   \let\MT@val\@tempb
3012 }

```

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

```

3013 \def\MT@get@font@#1#2#3#4#5#6{%
3014   \let\@tempb\@empty
3015   \def\MT@temp{#1/#2/#3/#4/#5}%
3016   \MT@get@axis{encoding}{#1}%
3017   \MT@get@axis{family} {#2}%
3018   \MT@get@axis{series} {#3}%
3019   \MT@get@axis{shape} {#4}%
3020   \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
3021   \MT@ifempty{#5}{%
3022     \MT@warn@axis@empty{size}{\string\normalsize}%
3023     \def\MT@val{*}%
3024   }{%
3025     \def\MT@val{#5}%
3026   }%
3027   \MT@get@size
3028 }

```

`\MT@get@axis`

```

3029 \def\MT@get@axis#1#2{%
3030   \def\MT@val{#2}%
3031   \MT@get@highlevel{#1}%
3032   \MT@ifempty\MT@val{%
3033     \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
3034     \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
3035   }\relax
3036   \expandafter@g@addto@macro\expandafter\@tempb\expandafter{\MT@val}/%
3037 }

```

`\MT@warn@axis@empty`

```

3038 \def\MT@warn@axis@empty#1#2{%
3039   \MT@warning{#1 axis is empty in font specification\MessageBreak
3040     ~\MT@temp'. Using ~#2' instead}%
3041 }

```

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

```

3042 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
3043   \MT@define@set@key@{encoding}{#1}%
3044   \MT@define@set@key@{family} {#1}%
3045   \MT@define@set@key@{series} {#1}%
3046   \MT@define@set@key@{shape} {#1}%
3047   \MT@define@set@key@size {#1}%
3048   \MT@define@set@key@font {#1}%
3049 }

```

`\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.

```

3050 \renewcommand*UseMicrotypeSet[2] [] {%
3051   \KV@sp@def\@tempa{#1}%
3052   \MT@ifempty\@tempa{%
3053     \MT@map@clist@c\MT@features{\MT@use@set{##1}{#2}}}%
3054   }{%
3055     \MT@map@clist@c\@tempa{%
3056       \KV@sp@def\@tempa{##1}%
3057       \MT@ifempty\@tempa\relax{%
3058         \MT@is@feature{activation of set `#2'}{%
3059           \MT@exp@one@n\MT@use@set
3060           {\csname MT@rbba@\@tempa\endcsname}{#2}%
3061         }%
3062       }%
3063     }%
3064   }%
3065 }

```

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

```

\MT@ex@setname 3066 \def\MT@use@set#1#2{%
\MT@tr@setname 3067   \KV@sp@def\@tempa{#2}%
3068   \MT@ifdefined@n@TF{MT@#1@set@\@tempa}{%
\MT@sp@setname 3069     \MT@xdef@n{MT@#1@setname}{\@tempa}%
\MT@kn@setname 3070   }{%
\MT@use@set 3071     \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
3072       \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
3073     }%
3074     \MT@error{%
3075       The \@nameuse{MT@abbr@#1} set `@\tempa' is undeclared.\MessageBreak
3076       Using set `@\nameuse{MT@#1@setname}' instead}{}%
3077   }%
3078 }

```

`\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.

```

3079 \renewcommand*DeclareMicrotypeSetDefault[2] [] {%
3080   \KV@sp@def\@tempa{#1}%
3081   \MT@ifempty\@tempa{%
3082     \MT@map@clist@c\MT@features{\MT@set@default@set{##1}{#2}}}%
3083   }{%
3084     \MT@map@clist@c\@tempa{%
3085       \KV@sp@def\@tempa{##1}%
3086       \MT@ifempty\@tempa\relax{%
3087         \MT@is@feature{declaration of default set `#2'}{%
3088           \MT@exp@one@n\MT@set@default@set
3089           {\csname MT@rbba@\@tempa\endcsname}{#2}%
3090         }%
3091       }%
3092     }%
3093   }%
3094 }

```

`\MT@default@pr@set`

```

\MT@default@ex@set 3095 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 3096   \KV@sp@def\@tempa{#2}%
3097   \MT@ifdefined@n@TF{MT@#1@set@\@tempa}{%
\MT@default@sp@set 3098     <debug>\MT@dinfo{1}{declaring default \@nameuse{MT@abbr@#1} set `@\tempa'}%
\MT@default@kn@set 3099     \MT@xdef@n{MT@default@#1@set}{\@tempa}%
\MT@set@default@set 3100   }{%
3101     \MT@error{%
3102       The \@nameuse{MT@abbr@#1} set `@\tempa' is not declared.\MessageBreak
3103       Cannot make it the default set. Using set\MessageBreak `all' instead}{}%
3104     \MT@xdef@n{MT@default@#1@set}{all}%
3105   }%

```

```
3106 }
```

14.3.2 Variants and aliases

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

```
3107 \let\MT@variants\@empty
3108 \def\DeclareMicrotypeVariants{%
3109   \ifstar
3110     \MT@DeclareVariants
3111     {\let\MT@variants\@empty\MT@DeclareVariants}%
3112 }
```

`\MT@DeclareVariants`

```
3113 \def\MT@DeclareVariants#1{%
3114   \MT@map@clist@n{#1}{%
3115     \KV@esp@def\@tempa{##1}%
3116     \@onelevel@sanitize\@tempa
3117     \xdef\MT@variants{\MT@variants{\@tempa}}%
3118   }%
3119 }
```

`\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.

```
3120 \renewcommand*\DeclareMicrotypeAlias[2]{%
3121   \edef\@tempa{\zap@space#1 \@empty}%
3122   \edef\@tempb{\zap@space#2 \@empty}%
3123   \@onelevel@sanitize\@tempb
3124   \MT@ifdefined@n@T{\MT@\@tempa @alias}{%
3125     \MT@warning{Alias font family '\@tempb' will override
3126       alias '\@nameuse{\MT@\@tempa @alias}'\MessageBreak
3127       for font family '\@tempa'}}%
3128   \MT@xdef@n{\MT@\@tempa @alias}{\@tempb}%
```

If we encounter this command while a font is being set up, we also set the alias for the current font so that if `\DeclareMicrotypeAlias` has been issued inside a configuration file, the configuration file for the alias font will be loaded, too.

```
3129 \MT@ifdefined@c@T\MT@family{%
3130 <debug>\MT@dinfo{1}{Activating alias font '\@tempb' for '\MT@family'}%
3131 \MT@glet\MT@familyalias\@tempb
3132 }%
3133 }
```

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

```
3134 \def\LoadMicrotypeFile#1{%
3135   \edef\@tempa{\zap@space#1 \@empty}%
3136   \@onelevel@sanitize\@tempa
3137   \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3138   \ifMT@inlist@
3139     \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
3140   \else
3141     \MT@xadd\MT@file@list{\@tempa,}%
3142     \MT@begin@catcodes
3143     \InputIfFileExists{mt-\@tempa.cfg}{%
3144       \edef\MT@curr@file{mt-\@tempa.cfg}%
3145       \MT@vinfo{... Loading configuration file \MT@curr@file}%
3146     }{%
3147       \MT@warning{... Configuration file mt-\@tempa.cfg\MessageBreak
3148         does not exist}%
3149     }%
3150     \MT@end@catcodes
3151   \fi
```



```

3152 }
3153 </package>
3154 </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@n1@setname` The optional argument may be used to disable selected ligatures only.

```

\MT@n1@ligatures 3155 <*pdftex-def|luatex-def>
3156 <pdftex-def>\MT@requires@pdftex5{
3157 \def\DisableLigatures{%
3158 \MT@begin@catcodes
3159 \MT@DisableLigatures
3160 }
3161 \newcommand*\MT@DisableLigatures[2] [] {%
3162 \MT@ifempty{#1}\relax{\gdef\MT@n1@ligatures{#1}}%
3163 \xdef\MT@active@features{\MT@active@features,nl}%
3164 \global\MT@noligaturestrue
3165 \MT@declare@sets{nl}{no ligatures}{#2}%
3166 \gdef\MT@n1@setname{no ligatures}%
3167 \MT@end@catcodes
3168 }
3169 <pdftex-def>}{
3170 </pdftex-def|luatex-def>

```

If pdfTeX is too old, we throw an error.

```

3171 <*pdftex-def|xetex-def>
3172 \renewcommand*\DisableLigatures[2] [] {%
3173 \MT@error{Disabling ligatures of a font is only possible\MessageBreak
3174 with pdftex version 1.30 or newer.\MessageBreak
3175 Ignoring \string\DisableLigatures}{%
3176 <pdftex-def> Upgrade
3177 <xetex-def> Use
3178 pdftex.}%
3179 }
3180 <pdftex-def>}{
3181 </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.

```

3182 <*package>
3183 \def\DeclareMicrotypeBabelHook#1#2{%
3184 \MT@map@clist@n{#1}{%
3185 \KV@sp@def\@tempa{##1}%
3186 \MT@gdef@n{\MT@babel@{\@tempa}{#2}}%
3187 }%
3188 }
3189 </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).

```
3190 ⟨*pdftex-def|xetex-def|luatex-def⟩
3191 \def\SetProtrusion{%
3192   \MT@begin@catcodes
3193   \MT@SetProtrusion
3194 }
```

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

```
\MT@pr@c@name 3195 \newcommand*\MT@SetProtrusion[3] [] {%
\MT@extra@context 3196   \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.

```
3197   \MT@set@named@keys{MT@pr@c}{#1}%
3198 ⟨debug⟩\MT@dinfn{1}{creating protrusion list `~\MT@pr@c@name'}%
3199   \def\MT@permutelist{pr@c}%
3200   \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⟩`, ...

```
3201   \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.

```
3202   \MT@gdefn{MT@pr@c@~\MT@pr@c@name}{#3}%
3203   \MT@end@catcodes
3204 }
3205 ⟨/pdftex-def|xetex-def|luatex-def⟩
```

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

```
3206 ⟨*pdftex-def|luatex-def⟩
3207 \def\SetExpansion{%
3208   \MT@begin@catcodes
3209   \MT@SetExpansion
3210 }
```

`\MT@SetExpansion`

```
\MT@ex@c@name 3211 \newcommand*\MT@SetExpansion[3] [] {%
\MT@extra@context 3212   \let\MT@extra@context\@empty
\MT@permutelist 3213   \MT@set@named@keys{MT@ex@c}{#1}%
3214   \MT@ifdefinedn@T{MT@ex@c@~\MT@ex@c@name @factor}{%
3215     \ifnum\c@name MT@ex@c@~\MT@ex@c@name @factor\endc@name > \@m
3216     \MT@warningn1{Expansion factor \number\@nameuse{MT@ex@c@~\MT@ex@c@name @factor}
3217       too large in list\MessageBreak `~\MT@ex@c@name'. Setting it to the
3218       maximum of 1000}%
3219     \MT@gletnc{MT@ex@c@~\MT@ex@c@name @factor}\@m
3220     \fi
3221   }%
3222 ⟨debug⟩\MT@dinfn{1}{creating expansion list `~\MT@ex@c@name'}%
3223   \def\MT@permutelist{ex@c}%
3224   \setkeys{MT@cfg}{#2}%
3225   \MT@permute
3226   \MT@gdefn{MT@ex@c@~\MT@ex@c@name}{#3}%
3227   \MT@end@catcodes
3228 }
```

`\SetTracking`

```
3229 \def\SetTracking{%
3230   \MT@begin@catcodes
3231   \MT@SetTracking
3232 }
```

`\MT@SetTracking` Third argument may be empty.

```

3233 \newcommand*\MT@SetTracking[3] [] {%
3234   \let\MT@extra@context\@empty
3235   \MT@set@named@keys{MT@tr@}{#1}%
3236   (debug)\MT@dinfo{1}{creating tracking list `~\MT@tr@{name}'}%
3237   \def\MT@permutelist{tr@}%
3238   \setkeys{MT@cfg}{#2}%
3239   \MT@permute
3240   \KV@sp@def@tempa{#3}%
3241   \MT@ifempty\@tempa\relax{%
3242     \MT@ifint\@tempa
3243     {\MT@xdef@n{MT@tr@{~\MT@tr@{name}}{\@tempa}}}%
3244     {\MT@warning{Value `~\@tempa' is not a number in\MessageBreak
3245       tracking set `~\MT@curr@set{name'}}}%
3246   \MT@end@catcodes
3247 }
3248 (/pdfTeX-def|luatex-def)

```

`\SetExtraSpacing`

```

3249 (*pdfTeX-def)
3250 \def\SetExtraSpacing{%
3251   \MT@begin@catcodes
3252   \MT@SetExtraSpacing
3253 }

```

`\MT@SetExtraSpacing`

```

\MT@sp@c@name 3254 \newcommand*\MT@SetExtraSpacing[3] [] {%
\MT@extra@context 3255   \let\MT@extra@context\@empty
\MT@permutelist 3256   \MT@set@named@keys{MT@sp@c}{#1}%
3257   (debug)\MT@dinfo{1}{creating spacing list `~\MT@sp@c{name}'}%
3258   \def\MT@permutelist{sp@c}%
3259   \setkeys{MT@cfg}{#2}%
3260   \MT@permute
3261   \MT@gdef@n{MT@sp@c@~\MT@sp@c{name}}{#3}%
3262   \MT@end@catcodes
3263 }

```

`\SetExtraKerning`

```

3264 \def\SetExtraKerning{%
3265   \MT@begin@catcodes
3266   \MT@SetExtraKerning
3267 }

```

`\MT@SetExtraKerning`

```

\MT@kn@c@name 3268 \newcommand*\MT@SetExtraKerning[3] [] {%
\MT@extra@context 3269   \let\MT@extra@context\@empty
\MT@permutelist 3270   \MT@set@named@keys{MT@kn@c}{#1}%
3271   (debug)\MT@dinfo{1}{creating kerning list `~\MT@kn@c{name}'}%
3272   \def\MT@permutelist{kn@c}%
3273   \setkeys{MT@cfg}{#2}%
3274   \MT@permute
3275   \MT@gdef@n{MT@kn@c@~\MT@kn@c{name}}{#3}%
3276   \MT@end@catcodes
3277 }
3278 (/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.

```

3279 (*package)
3280 \def\MT@set@named@keys#1#2{%
3281   \def\x##1name=##2,##3\@nil{%
3282     \setkeys{#1}{name=##2}%
3283     \gdef\MT@options{##1##3}%
3284     \MT@rem@from@clist{name=}\MT@options

```

```

3285 }%
3286 \x#2,name=,\@nil
3287 \@expandtwoargs\setkeys{#1}\MT@options
3288 }

```

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

```

3289 \def\MT@define@code@key#1#2{%
3290   \define@key{MT@#2}{#1}[]{}%
3291   \@tempcnta=\@ne
3292   \MT@map@clist@n{##1}%
3293   \KV@sp@def\MT@val{###1}%

```

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

```

3294   \MT@get@highlevel{#1}%
3295   \MT@edef@n{MT@temp#1\the\@tempcnta}\MT@val}%
3296   \advance\@tempcnta \@ne
3297 }%
3298 }%
3299 }

```

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

```

3300 \def\MT@define@code@key@family#1{%
3301   \define@key{MT@#1}{family}[]{}%
3302   \@tempcnta=\@ne
3303   \MT@map@clist@n{##1}%
3304   \KV@sp@def\MT@val{###1}%
3305   \MT@get@highlevel{family}%
3306   \ifMT@fontspec
3307     \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
3308     \fi
3309     \MT@edef@n{MT@tempfamily\the\@tempcnta}\MT@val}%
3310     \advance\@tempcnta \@ne
3311 }%
3312 }%
3313 }

```

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

```

3314 \def\MT@define@code@key@size#1{%
3315   \define@key{MT@#1}{size}[]{}%
3316   \MT@map@clist@n{##1}%
3317   \KV@sp@def\MT@val{###1}%
3318   \expandafter\MT@get@range\MT@val--\@nil
3319   \ifx\MT@val\relax \else
3320     \MT@exp@cs\MT@xadd{MT@tempsize}%
3321     {{{\MT@lower}\MT@upper}\MT@curr@set@name}}}%
3322   \fi
3323 }%
3324 }%
3325 }

```

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

```

3326 \def\MT@define@code@key@font#1{%
3327   \define@key{MT@#1}{font}[]{}%
3328   \MT@map@clist@n{##1}%
3329   \KV@sp@def\MT@val{###1}%
3330   \MT@ifstreq\MT@val*{\def\MT@val{*/**/*/*}}\relax
3331   \expandafter\MT@get@font@and@size\MT@val///// \@nil
3332   \ifMT@fontspec
3333     \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
3334     \fi
3335     \MT@xdef@n{MT@MT@permutelist @\@tempb\MT@extra@context}%
3336     {\csname MT@MT@permutelist @name\endcsname}%
3337     <debug>\MT@dinfo@n{1}{initialising: use list for font \@tempb=\MT@val
3338     <debug> \ifx\MT@extra@context\@empty\else\MessageBreak

```

```

3339 (debug) (context: \MT@extra@context)\fi}%
3340 \MT@exp@cs\MT@xaddb
3341 {MT@\MT@permutelist @\@tempb\MT@extra@context @sizes}%
3342 {{{\MT@eval}{\m@ne}{\MT@curr@set@name}}}%
3343 }%
3344 }%
3345 }

```

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

```

3346 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\nil{%
3347 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
3348 }

3349 \MT@define@code@key{encoding}{cfg}
3350 \MT@define@code@key@family {cfg}
3351 \MT@define@code@key{series} {cfg}
3352 \MT@define@code@key{shape} {cfg}
3353 \MT@define@code@key@size {cfg}
3354 \MT@define@code@key@font {cfg}

```

`\MT@define@opt@key`

```

3355 \def\MT@define@opt@key#1#2{%
3356 \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax{%
3357 \MT@xdef@n{MT@#1@c@MT@curr@set@name @#2}{##1}}}%
3358 }

```

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

```

3359 \newcount\MT@listname@count
3360 \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`).

```

3361 \define@key{MT@#1@c}{name}[]{%
3362 \MT@ifempty{##1}{%
3363 \MT@ifdefined@n@TF{MT@#1@c@MT@curr@file/\the\inputlineno}{%
3364 \global\advance\MT@listname@count@one
3365 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
3366 (\number\MT@listname@count)}%
3367 }{%
3368 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
3369 }%
3370 }{%
3371 \MT@edef@n{MT@#1@c@name}{##1}%
3372 \MT@ifdefined@n@T{MT@#1@c@csname MT@#1@c@name\endcsname}{%
3373 \MT@warning{Redefining \nameuse{MT@abbr@#1} list `\'nameuse{MT@#1@c@name}'}%
3374 }%
3375 }%
3376 \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
3377 }%
3378 \MT@define@opt@key{#1}{load}%
3379 \MT@define@opt@key{#1}{factor}%
3380 \MT@define@opt@key{#1}{preset}%
3381 \MT@define@opt@key{#1}{inputenc}%

```

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

```

3382 \define@key{MT@#1@c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
3383 }
3384 (/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.

```

3385 (*pdfTeX-def|luatex-def)

```

```

3386 <pdfTEX-def>\MT@requires@pdfTEX7{
3387 \define@key{MT@ex@c}{context}[]{}%
3388 \MT@ifempty{#1}\relax{%
3389 \MT@gllet\MT@copy@font\MT@copy@font@
3390 \def\MT@extra@context{#1}%
3391 }%
3392 }
3393 \MT@addto@setup{%
3394 \define@key{MT@ex@c}{context}[]{}%
3395 \ifx\MT@copy@font\MT@copy@font@
3396 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
3397 \else
3398 \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
3399 Ignoring `context' key\on@line}%
3400 {Either move the settings inside the preamble,\MessageBreak
3401 or load the package with the `copyfonts' option.}%
3402 \fi
3403 }%
3404 }

```

Protrusion contexts *may* 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.

```

3405 \define@key{MT@pr@c}{context}[]{}%
3406 \MT@ifempty{#1}\relax{%
3407 \MT@gllet\MT@copy@font\MT@copy@font@
3408 \def\MT@extra@context{#1}%
3409 }%
3410 }
3411 \MT@addto@setup{%
3412 \define@key{MT@pr@c}{context}[]{}%
3413 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
3414 \ifx\MT@copy@font\MT@copy@font@\else
3415 \MT@warning@n1{If protrusion contexts don't work as expected,
3416 \MessageBreak load the package with the `copyfonts' option}%
3417 \fi
3418 }%
3419 }
3420 </pdfTEX-def|luatEX-def>
3421 <*pdfTEX-def>
3422 {}{
3423 \define@key{MT@ex@c}{context}[]{}%
3424 \MT@error{Expansion contexts only work with pdfTEX 1.40.4\MessageBreak
3425 or later. Ignoring `context' key\on@line}%
3426 {Upgrade pdfTEX.}%
3427 }
3428 </pdfTEX-def>
3429 <*pdfTEX-def|xetEX-def>
3430 \define@key{MT@pr@c}{context}[]{}%
3431 \MT@error{Protrusion contexts only work with pdfTEX
3432 <pdfTEX-def> 1.40.4\MessageBreak or later.
3433 <xetEX-def> \MessageBreak or luatEX.
3434 Ignoring `context' key\on@line}%
3435 <pdfTEX-def> {Upgrade pdfTEX.}%
3436 <xetEX-def> {Use pdfTEX or luatEX.}%
3437 }
3438 </pdfTEX-def|xetEX-def>
3439 <pdfTEX-def>

```

\MT@warn@nodim

```

3440 <*package>
3441 \def\MT@warn@nodim#1{%
3442   \MT@warning{'\@tempa' is not a dimension.\MessageBreak
3443     Ignoring it and setting values relative to\MessageBreak #1}%
3444 }
3445 </package>

```

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

```

3446 <*pdfTeX-def|xetex-def|luatex-def>
3447 \define@key{MT@pr@c}{unit}[character]{%
3448   \MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
3449   \def\@tempa{#1}%
3450   \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.

```

3451   \MT@ifdimen\@tempa
3452     {\MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
3453     {\MT@warn@nodim{character widths}}%
3454   }%
3455 }
3456 </pdfTeX-def|xetex-def|luatex-def>

```

Tracking may only be relative to a dimension.

```

3457 <*pdfTeX-def|luatex-def>
3458 \define@key{MT@tr@c}{unit}[1em]{%
3459   \MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@empty
3460   \def\@tempa{#1}%
3461   \MT@ifdimen\@tempa
3462     {\MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@tempa}%
3463     {\MT@warn@nodim{1em}}%
3464   \MT@gdefn{MT@tr@c@MT@curr@set@name @unit}{1em}}%
3465 }
3466 </pdfTeX-def|luatex-def>

```

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

```

3467 <*pdfTeX-def>
3468 \MT@map@clist@{sp,kn}{%
3469   \define@key{MT@#1@c}{unit}[space]{%
3470     \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@empty
3471     \def\@tempa{##1}%
3472     \MT@ifstreq\@tempa{character}\relax{%
3473       \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\m@ne
3474       \MT@ifstreq\@tempa{space}\relax{%
3475         \MT@ifdimen\@tempa
3476           {\MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@tempa}%
3477           {\MT@warn@nodim{width of space}}%
3478       }%
3479     }%
3480   }%
3481 }
3482 </pdfTeX-def>

```

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

```

3483 <*pdfTeX-def|luatex-def>
3484 \MT@map@clist@{stretch,shrink,step}{%
3485   \define@key{MT@ex@c}{#1}[]{%
3486     \MT@ifempty{##1}\relax{%
3487       \MT@ifint{##1}{%

```

A space terminates the number.

```

3488     \MT@gdefn{MT@ex@c@MT@curr@set@name @#1}{##1 }%
3489   }{%
3490     \MT@warning{%
3491       Value `##1' for option `#1' is not a number.\MessageBreak

```

```

3492     Ignoring it}%
3493   }%
3494 }%
3495 }%
3496 }
3497 \define@key{MT@ex@c}{auto}[true]{%
3498   \def\@tempa{#1}%
3499   \csname if\@tempa\endcsname

Don't use autoexpand for pdfTeX version older than 1.20.
3500 \pdfTeX-def \MT@requires@pdftex4{%
3501   \MT@gdef@n{MT@ex@c@MT@curr@set@name @auto}{autoexpand}%
3502 \*pdfTeX-def
3503 }{%
3504   \MT@warning{pdfTeX too old for automatic font expansion}%
3505 }
3506 \pdfTeX-def
3507 \else
3508 \pdfTeX-def \MT@requires@pdftex4{%
3509   \MT@glet@nc{MT@ex@c@MT@curr@set@name @auto}\@empty
3510 \pdfTeX-def } \relax
3511 \fi
3512 }

```

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

```

3513 \MT@define@opt@key{tr}{spacing}
3514 \MT@define@opt@key{tr}{outerspacing}
3515 \MT@define@opt@key{tr}{outerkerning}

```

Which ligatures should be disabled?

```

3516 \define@key{MT@tr@c}{noligatures}[]%
3517   {\MT@xdef@n{MT@tr@c@MT@curr@set@name @noligatures}{#1}}
3518 \define@key{MT@tr@c}{outer spacing}[]{\setkeys{MT@tr@c}{outerspacing={#1}}}
3519 \define@key{MT@tr@c}{outer kerning}[]{\setkeys{MT@tr@c}{outerkerning={#1}}}
3520 \define@key{MT@tr@c}{no ligatures}[]{\setkeys{MT@tr@c}{noligatures={#1}}}
3521 \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.

```

3522 \*package
3523 \renewcommand*\DeclareCharacterInheritance[1] [] {%
3524   \let\MT@extra@context\@empty
3525   \let\MT@extra@inputenc\@undefined
3526   \let\MT@inh@feat\@empty
3527   \setkeys{MT@inh@}{#1}%
3528   \MT@begin@catcodes
3529   \MT@set@inh@list
3530 }

```

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

```

3531 \def\MT@set@inh@list#1#2{%
3532   \MT@ifempty\MT@inh@feat{%

```



```

3533 \MT@map@clist@c\MT@features{\MT@declare@char@inh{##1}{##2}}%
3534 }{%
3535 \MT@map@clist@c\MT@inh@feat{%
3536 \KV@esp@def\@tempa{##1}%
3537 \MT@ifempty\@tempa\relax{%
3538 \MT@exp@one@n\MT@declare@char@inh
3539 {\csname MT@rbba@\@tempa\endcsname}{##1}{##2}%
3540 }%
3541 }}%
3542 }%
3543 \MT@end@catcodes
3544 }

```

The keys for the optional argument.

```

3545 \MT@map@clist@c\MT@features@long{%
3546 \define@key{MT@inh@}{#1}[]{\edef\MT@inh@feat{\MT@inh@feat#1,}}
3547 \define@key{MT@inh@}{inputenc}{\def\MT@extra@inputenc{#1}}

```

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

```

3548 \def\MT@declare@char@inh#1#2#3{%
3549 \MT@edef@n{MT@#1@inh@name}%
3550 {\MT@curr@file/\the@inputlineno (\@nameuse{MT@abbr@#1})}%
3551 \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
3552 \MT@ifdefined@c@T\MT@extra@inputenc{%
3553 \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
3554 <debug>\MT@dinfo{1}{creating inheritance list \@nameuse{MT@#1@inh@name}'}%
3555 \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
3556 \def\MT@permutelist{#1@inh}%
3557 \setkeys{MT@inh}{#2}%
3558 \MT@permute
3559 }

```

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

```

3560 \MT@define@code@key{encoding}{inh}
3561 \MT@define@code@key@family {inh}
3562 \MT@define@code@key{series} {inh}
3563 \MT@define@code@key{shape} {inh}
3564 \MT@define@code@key@size {inh}
3565 \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`).

```

3566 \def\MT@inh@do#1,{%
3567 \ifx\relax#1\@empty \else
3568 \MT@inh@split #1==\relax
3569 \expandafter\MT@inh@do
3570 \fi
3571 }

```

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

```

3572 </package>
3573 <*pdfTeX-def|xetex-def|luatex-def>
3574 \def\MT@inh@split#1=#2=#3\relax{%
3575 \def\@tempa{#1}%
3576 \ifx\@tempa\@empty \else
3577 \MT@get@slot
3578 <pdfTeX-def|luatex-def> \ifnum\MT@char > \m@ne

```

```

3579 <xetex-def> \ifx\MT@char\@empty\else
3580 \let\MT@val\MT@char
3581 \MT@map@clist@n{#2}{%
3582 \def\@tempa{##1}%
3583 \ifx\@tempa\@empty \else
3584 \MT@get@slot
3585 <pdfTEX-def|lUATeX-def> \ifnum\MT@char > \m@ne
3586 <xetex-def> \ifx\MT@char\@empty\else
3587 \MT@exp@cs\MT@xadd\MT@inh\MT@listname @\MT@val @{{\MT@char}}%
3588 \fi
3589 \fi
3590 }%
3591 <debug>\MT@dinfo@n1{2}{children of #1 (\MT@val):
3592 <debug> \@nameuse{MT@inh\MT@listname @\MT@val @}}%
3593 \fi
3594 \fi
3595 }
3596 </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. 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.

```

3597 <*package>
3598 \def\MT@permute{%
3599 \let\MT@cnt@encoding\@ne
3600 \MT@permute@

  Undefine commands for the next round.
3601 \MT@map@tlist@n{{encoding}{family}{series}{shape}}\MT@permute@reset
3602 \MT@glet\MT@tempsize\@undefined
3603 }
3604 \def\MT@permute@{%
3605 \let\MT@cnt@family\@ne
3606 \MT@permute@@
3607 \MT@increment\MT@cnt@encoding
3608 \MT@ifdefined@n@T{MT@tempencoding\MT@cnt@encoding}%
3609 \MT@permute@
3610 }
3611 \def\MT@permute@@{%
3612 \let\MT@cnt@series\@ne
3613 \MT@permute@@@
3614 \MT@increment\MT@cnt@family
3615 \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
3616 \MT@permute@@@
3617 }
3618 \def\MT@permute@@@{%
3619 \let\MT@cnt@shape\@ne
3620 \MT@permute@@@@
3621 \MT@increment\MT@cnt@series
3622 \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
3623 \MT@permute@@@@
3624 }
3625 \def\MT@permute@@@@{%
3626 \MT@permute@@@@@
3627 \MT@increment\MT@cnt@shape
3628 \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
3629 \MT@permute@@@@@
3630 }

```

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

```

3631 \def\MT@permute@{
3632   \MT@permute@define{encoding}%
3633   \ifMT@document
3634     \ifx\MT@tempencoding\empty \else
3635       \MT@ifdefined@n@TF{TF\MT@tempencoding}\relax
3636       {\expandafter\expandafter\expandafter\@gobble}%
3637     \fi
3638   \fi
3639   \MT@permute@
3640 }

```

`\MT@permute@`

```

3641 \def\MT@permute@{
3642   \MT@permute@define{family}%
3643   \MT@permute@define{series}%
3644   \MT@permute@define{shape}%
3645   \edef\@tempa{\MT@tempencoding
3646     /\MT@tempfamily
3647     /\MT@tempseries
3648     /\MT@tempshape
3649     /\MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

3650 \MT@ifstreq\@tempa{////}\relax%
3651 \ifx\MT@tempencoding\empty
3652   \MT@warning{
3653     You have to specify an encoding for\MessageBreak
3654     \@nameuse{MT@abbr@MT@permutelist} list
3655     ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
3656     Ignoring it}%
3657   \else
3658     \MT@ifdefined@c@TF\MT@tempsize%

```

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

```

3659   \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%
3660     \MT@map@tlist@c\MT@tempsize\MT@check@rlist
3661   }%
3662   \MT@exp@cs\MT@xaddb
3663   {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
3664   \MT@tempsize
3665   <debug>\MT@dinfo@n1{1}{initialising: use list for font \@tempa,\MessageBreak
3666   <debug>     sizes: \csname MT@MT@permutelist @\@tempa\MT@extra@context
3667   <debug>     @sizes\endcsname}%
3668   }{%

```

Only one list can apply to a given combination.

```

3669   \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context}{%
3670     \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
3671       ~\@nameuse{MT@MT@permutelist @name}' will override list\MessageBreak
3672       ~\@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
3673       for font ~\@tempa'}%
3674   }%
3675   <debug>\MT@dinfo@n1{1}{initialising: use list for font \@tempa
3676   <debug>     \ifx\MT@extra@context\empty\else\MessageBreak
3677   <debug>     (context: \MT@extra@context)\fi}%
3678   }%
3679   \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}{%
3680     {\csname MT@MT@permutelist @name\endcsname}%
3681   \fi
3682   }%
3683 }

```

```

\MT@permute@define    Define the commands.
3684 \def\MT@permute@define#1{%
3685   \@tempcnta=\csname MT@cnt@#1\endcsname\relax
3686   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3687   {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
3688   {\MT@let@nc{MT@temp#1}\empty}%
3689 }

\MT@permute@reset    Reset the commands.
3690 \def\MT@permute@reset#1{%
3691   \@tempcnta=\@ne
3692   \MT@loop
3693   \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
3694   \advance\@tempcnta\@ne
3695   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3696   \iftrue
3697   \iffalse
3698   \MT@repeat
3699 }

\MT@check@rlist    For every new range item in \MT@tempsize, check whether it overlaps with ranges
                   in the existing list.
3700 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

\MT@check@rlist@    Define the current new range and ...
3701 \def\MT@check@rlist@#1#2#3{%
3702   \def\@tempb{#1}%
3703   \def\@tempc{#2}%
3704   \MT@iffalse
3705   \MT@exp@cs\MT@map@tlist@c
3706   {MT\MT@permutelist @\@tempa\MT@extra@context @sizes}%
3707   \MT@check@range
3708 }

\MT@check@range    ... recurse through the list of existing ranges.
3709 \def\MT@check@range#1{\expandafter\MT@check@range@ #1}

\MT@check@range@    \@tempb and \@tempc are lower resp. upper bound of the new range, <#2> and <#3>
                   those of the existing range.
3710 \def\MT@check@range@#1#2#3{%
3711   \MT@ifdim{#2}=\m@ne{%
3712     \MT@ifdim\@tempc=\m@ne{%


- Both items are simple sizes.


3713     \MT@ifdim\@tempb={#1}\MT@if@true\relax
3714     }{%


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


3715     \MT@ifdim\@tempb>{#1}\relax{%
3716       \MT@ifdim\@tempc>{#1}{%
3717         \MT@if@true
3718         \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
3719         }\relax
3720       }%
3721     }%
3722   }{%


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


3724     \MT@ifdim\@tempb<{#2}{%
3725       \MT@ifdim\@tempb<{#1}\relax\MT@if@true

```

```
3726     }\relax
3727     }{%
```

- Both items are ranges.

```
3728     \MT@ifdim\@tempb<{#2}{%
3729     \MT@ifdim\@tempc>{#1}{%
3730     \MT@iftrue
3731     \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
3732     }\relax
3733     }\relax
3734     }%
3735     }%
3736     \ifMT@if@
3737     \MT@warning{\@nameuse{MT@abbr@\MT@permutelist} list
3738     ~\@nameuse{MT@\MT@permutelist @name}' will override\MessageBreak
3739     list ~#3' for font \@tempa,\MessageBreak size \@tempb}%
```

If we've already found a conflict with this item, we can skip the rest of the list.

```
3740     \expandafter\MT@tlist@break
3741     \fi
3742 }
```

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 3743 \newif\ifMT@opt@expansion
\ifMT@opt@DVI 3744 \newif\ifMT@opt@auto
3745 \newif\ifMT@opt@DVI
```

`\MT@optwarn@admissible` Some warnings.

```
3746 \def\MT@optwarn@admissible#1#2{%
3747 \MT@warning@nl{~#1' is not an admissible value for option\MessageBreak
3748 ~#2'. Assuming ~false'}%
3749 }
```

`\MT@optwarn@nan`

```
3750 </package>
3751 <*package|letterspace>
3752 <plain>\MT@requires@latex1{
3753 \def\MT@optwarn@nan#1#2{%
3754 \MT@warning@nl{Value ~#1' for option ~#2' is not a\MessageBreak number.
3755 Using default value of \number\@nameuse{MT@#2@default}}%
3756 }
3757 <plain>}\relax
3758 </package|letterspace>
3759 <*package>
```

`\MT@opt@def@set`

```
3760 \def\MT@opt@def@set#1{%
3761 \MT@ifdefined@n@TF{MT@\@tempb @set@\MT@val}{%
3762 \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
3763 }{%
3764 \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
3765 \MT@warning@nl{The #1 set ~\MT@val' is undeclared.\MessageBreak
3766 Using set ~\@nameuse{MT@\@tempb @setname}' instead}%
3767 }%
3768 }
```

expansion and protrusion may be true, false, compatibility, nocompatibility and/or a *<set name>*.

```

3769 \MT@map@clist@n{protrusion,expansion}{%
3770   \define@key{MT}{#1}[true]{%
3771     \csname MT@opt@#1true\endcsname
3772     \MT@map@clist@n{##1}{%
3773       \KV@sp@def\MT@val{###1}%
3774       \MT@ifempty\MT@val\relax{%
3775         \csname MT@#1true\endcsname
3776         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
3777         \MT@ifstreq\MT@val{true}\relax
3778         {%
3779           \MT@ifstreq\MT@val{false}{%
3780             \csname MT@#1false\endcsname
3781           }{%
3782             \MT@ifstreq\MT@val{compatibility}{%
3783               \MT@let@nc{MT@\@tempb @level}\@one
3784             }{%
3785               \MT@ifstreq\MT@val{nocompatibility}{%
3786                 \MT@let@nc{MT@\@tempb @level}\tw@
3787               }{%

```

If everything failed, it should be a set name.

```

3788     \MT@opt@def@set{#1}%
3789   }%
3790 }%
3791 }%
3792 }%
3793 }%
3794 }%
3795 }%
3796 }

```

activate is a shortcut for protrusion and expansion.

```

3797 \define@key{MT}{activate}[true]{%
3798   \setkeys{MT}{protrusion=#1}%
3799   \setkeys{MT}{expansion=#1}%
3800 }

```

spacing, kerning and tracking do not have a compatibility level.

```

3801 \MT@map@clist@n{spacing,kerning,tracking}{%
3802   \define@key{MT}{#1}[true]{%
3803     \MT@map@clist@n{##1}{%
3804       \KV@sp@def\MT@val{###1}%
3805       \MT@ifempty\MT@val\relax{%
3806         \csname MT@#1true\endcsname
3807         \MT@ifstreq\MT@val{true}\relax
3808         {%
3809           \MT@ifstreq\MT@val{false}{%
3810             \csname MT@#1false\endcsname
3811           }{%
3812             \edef\@tempb{\csname MT@rbba@#1\endcsname}%
3813             \MT@opt@def@set{#1}%
3814           }%
3815         }%
3816       }%
3817     }%
3818   }%
3819 }

```

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

```

3820 \def\MT@def@bool@opt#1#2{%
3821   \define@key{MT}{#1}[true]{%

```

```

3822 \def\@tempa{##1}%
3823 \MT@ifstreq\@tempa{true}\relax{%
3824 \MT@ifstreq\@tempa{false}\relax{%
3825 \MT@optwarn@admissible{##1}{#1}%
3826 \def\@tempa{false}%
3827 }%
3828 }%
3829 #2%
3830 }%
3831 }

```

Boolean options that only set the switch.

```

3832 \MT@map@clist@n{draft,selected,babel}{%
3833 \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
3834 \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.

```

3835 </package>
3836 <*pdfTeX-def|luatex-def|xetex-def>
3837 \MT@def@bool@opt{DVIoutput}{%
3838 \csname if\@tempa\endcsname
3839 <*pdfTeX-def|luatex-def>
3840 \ifnum\pdfoutput>\z@ \MT@opt@DVITrue \fi
3841 \pdfoutput\z@
3842 \else
3843 \ifnum\pdfoutput<\@ne \MT@opt@DVITrue \fi
3844 \pdfoutput\@ne
3845 </pdfTeX-def|luatex-def>
3846 <xetex-def> \MT@warning@n1{Ignoring `DVIoutput' option}%
3847 \fi
3848 }
3849 </pdfTeX-def|luatex-def|xetex-def>

```

Setting the `defersetup` option to false will restore the old behaviour, where the setup took place at the time when the package was loaded. This is undocumented, since I would like to learn about the cases where this is necessary.

The only problem with the new deferred setup I can think of is when a box is being constructed inside the preamble and this box contains a font that is not loaded before the box is being used.

```

3850 <*package>
3851 \MT@def@bool@opt{defersetup}{%
3852 \csname if\@tempa\endcsname \else
3853 \AtEndOfPackage{%
3854 \MT@setup@
3855 \let\MT@setup@\empty
3856 \let\MT@addto@setup@\firstofone
3857 }%
3858 \fi
3859 }
3860 </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.

```

3861 <*pdfTeX-def|luatex-def>
3862 <pdfTeX-def>\MT@requires@pdfTeX7{
3863 \MT@def@bool@opt{copyfonts}{%
3864 \csname if\@tempa\endcsname
3865 \MT@gllet\MT@copy@font\MT@copy@font@
3866 \else

```

```

3867 \MT@gllet\MT@copy@font\relax
3868 \fi
3869 }
3870 <pdfTex-def>{
3871 </pdfTex-def|luatex-def>
3872 <*pdfTex-def|xetex-def>
3873 \MT@def@bool@opt{copyfonts}{%
3874 \csname if\@tempa\endcsname
3875 \MT@error
3876 <pdfTex-def> {The pdfTex version you are using is too old\MessageBreak
3877 <pdfTex-def> to use the `copyfonts' option}{Upgrade pdfTex.}%
3878 <xetex-def> {The `copyfonts' option does not work with xetex}
3879 <xetex-def> {Use pdfTex or luatex instead.}%
3880 \fi
3881 }
3882 <pdfTex-def>
3883 </pdfTex-def|xetex-def>

```

final is the opposite to draft.

```

3884 <*package>
3885 \MT@def@bool@opt{final}{%
3886 \csname if\@tempa\endcsname
3887 \MT@draftfalse
3888 \else
3889 \MT@drafttrue
3890 \fi
3891 }

```

For verbose output, we redefine \MT@vinfo.

```

3892 \define@key{MT}{verbose}[true]{%
3893 \let\MT@vinfo\MT@info@n1
3894 \def\@tempa{#1}%
3895 \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

3896 \MT@ifstreq\@tempa{errors}{%
3897 \let\MT@warning \MT@warn@err
3898 \let\MT@warning@n1\MT@warn@err
3899 }{%
3900 \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

3901 \MT@ifstreq\@tempa{silent}{%
3902 \let\MT@warning \MT@info
3903 \let\MT@warning@n1\MT@info@n1
3904 }{%
3905 \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
3906 }%
3907 }%
3908 }%
3909 }
3910 </package>

```

Options with numerical keys: factor, stretch, shrink, step, letterspace.

```

3911 <*package|letterspace>
3912 <plain>\MT@requires@l@texl{
3913 \MT@map@clist@n{%
3914 <package> stretch,shrink,step,%
3915 letterspace}{%
3916 \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
3917 \def\@tempa{##1 }%

```

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

```

3918 \MT@ifint\@tempa
3919 {\MT@edef@n{MT@#1}{\@tempa}}%

```



```

3920     {\MT@optwarn@nan{##1}{#1}}%
3921   }%
3922 }
3923 {plain}}\relax
3924 {/package|letterspace}

```

factor will define the protrusion factor only.

```

3925 {*package}
3926 \define@key{MT}{factor}[\MT@factor@default]{%
3927   \def\@tempa{#1 }%
3928   \MT@ifint\@tempa
3929     {\edef\MT@pr@factor{\@tempa}}
3930     {\MT@optwarn@nan{#1}{factor}}%
3931 }

```

Unit for protrusion codes.

```

3932 \define@key{MT}{unit}[character]{%
3933   \def\@tempa{#1}%
3934   \MT@ifstreq\@tempa{character}\relax{%
3935     \MT@ifdimen\@tempa
3936     {\let\MT@pr@unit\@tempa}%
3937     {\MT@warning@nl{"\@tempa' is not a dimension.\MessageBreak
3938       Ignoring it and setting values relative to\MessageBreak
3939       character widths}}%
3940   }%
3941 }

```

14.4.2 Loading the definition file

`\MT@endinput` Abort if no capable engine found.

```

3942 \let\MT@endinput\relax
3943 \ifx\MT@engine\relax
3944   \MT@warning@nl{You don't seem to be using either pdftex, luatex, or xetex.\MessageBreak
3945     "\MT@MT' only works with these engines.\MessageBreak
3946     I will quit now.}
3947   \MT@clear@options
3948 \else
3949   \input{microtype-\MT@engine tex.def}
3950 \fi
3951 \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 \TeX systems have switched to the pdf \TeX engine even for DVI output, so that the user might not even be aware of the fact that she's running pdf \TeX .)

```

3952 \MT@protrusiontrue
3953 {/package}
3954 {*pdftex-def|luatex-def}
3955 \ifnum\pdfoutput<\@ne \else

```

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

```

3956 {pdftex-def} \MT@requires@pdftex4{
3957   \MT@expansiontrue
3958   \MT@autottrue
3959 {pdftex-def} }\relax
3960 \fi
3961 {/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
`\MT@get@config` to define a no-op for the regular option processing later.

```

3962 (*package)
3963 \define@key{MT}{config}[]{\relax}
3964 \def\MT@get@config#1config=#2,#3\@nil{%
3965   \MT@ifempty{#2}%
3966   {\def\MT@config@file{\MT@MT.cfg}}%
3967   {\def\MT@config@file{#2.cfg}}%
3968 }
3969 \expandafter\expandafter\expandafter\MT@get@config
3970 \csname opt@\currname.\@currxt\endcsname,config=\@nil

```

Load the file.

```

3971 \IfFileExists{\MT@config@file}{%
3972   \MT@info@nl{Loading configuration file \MT@config@file}%
3973   \MT@begin@catcodes
3974   \let\MT@begin@catcodes\relax
3975   \let\MT@end@catcodes\relax
3976   \let\MT@curr@file\MT@config@file
3977   \input{\MT@config@file}%
3978   \endgroup
3979 }{\MT@warning@nl{%
3980   Could not find configuration file `~\MT@config@file'!\MessageBreak
3981   This will almost certainly cause undesired results.\MessageBreak
3982   Please fix your installation}%
3983 }

```

`\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).

```

3984 \def\MT@check@active@set#1{%
3985   \MT@ifdefined@n@TF{MT@#1@setname}{%
3986     \MT@info@nl{Using \@nameuse{MT@abbr@#1} set `~\@nameuse{MT@#1@setname}'}%
3987   }{%
3988     \MT@ifdefined@n@TF{MT@default@#1@set}{%
3989       \MT@gl@et@nn{MT@#1@setname}{MT@default@#1@set}%
3990       \MT@info@nl{Using default \@nameuse{MT@abbr@#1} set `~\@nameuse{MT@#1@setname}'}%
3991     }{%

```

If no default font set has been declared in the main configuration file, we use the (empty, non-existent) set '@', and issue a warning.

```

3992   \MT@gdef@n{MT@#1@setname}{@}%
3993   \MT@warning@nl{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
3994     \MessageBreak Using empty set}%
3995   }%
3996 }%
3997 }

```

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.

```
3998 \MT@ifdefined@c@T\MicroType@Hook{\MT@warning{%
3999   Command \string\MicroType@Hook\space is deprecated.\MessageBreak
4000   Use \string\Microtype@Hook\space instead}\MicroType@Hook}
4001 \MT@ifdefined@c@T\Microtype@Hook\Microtype@Hook
```

14.4.5 Changing options later

`\microtypesetup`
`\MT@define@optionX`

Inside the preamble, `\microtypesetup` accepts the same options as the package (unless `defersetup=false`). In the document body, it accepts the options: protrusion, expansion, activate, tracking, spacing and kerning. Specifying font sets is not allowed.

```
4002 \def\microtypesetup{\setkeys{MT}}
4003 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
4004 /package
4005 (*pdfTeX-def|luatex-def|xetex-def)
4006 \def\MT@define@optionX#1#2{%
4007   \define@key{MTX}{#1}[true]{%
4008     \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4009     \MT@map@clist@n{##1}{%
4010       \KV@sp@def\MT@val{###1}%
4011       \MT@ifempty\MT@val\relax{%
4012         \@tempcnta=\m@ne
4013         \MT@ifstreq\MT@val{true}{%
```

Enabling micro-typography in the middle of the document is not allowed if it has been disabled in the package options since fonts might already have been loaded and hence wouldn't be set up.

```
4014   \MT@checksetup{#1}{%
4015     \@tempcnta=\csname MT@\@tempb @level\endcsname
4016     \MT@vinfo{Enabling #1
4017       (level \number\csname MT@\@tempb @level\endcsname)\on@line}%
4018     }%
4019   }{%
4020     \MT@ifstreq\MT@val{false}{%
4021       \@tempcnta=\z@
4022       \MT@vinfo{Disabling #1\on@line}%
4023     }{%
4024       \MT@ifstreq\MT@val{compatibility}{%
4025         \MT@checksetup{#1}{%
4026           \@tempcnta=\@ne
4027           \MT@let@nc{MT@\@tempb @level}\@ne
4028           \MT@vinfo{Setting #1 to level 1\on@line}%
4029         }%
4030       }{%
4031         \MT@ifstreq\MT@val{nocompatibility}{%
4032           \MT@checksetup{#1}{%
```

```

4033         \@tempcnta=\tw@
4034         \MT@let@nc{MT@\@tempb @level}\tw@
4035         \MT@vinfo{Setting #1 to level 2\on@line}%
4036     }%
4037     }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
4038         {Use any of `~true', `~false', `~compatibility' or
4039         `~nocompatibility'.}%
4040     }%
4041 }%
4042 }%
4043 }%
4044 \ifnum\@tempcnta>\m@ne
4045     #2\@tempcnta\relax
4046 \fi
4047 }%
4048 }%
4049 }%
4050 }

```

`\MT@checksetup` Test whether the feature wasn't disabled in the package options.

```

4051 \def\MT@checksetup#1{%
4052     \csname ifMT@#1\endcsname
4053     \expandafter\@firstofone
4054     \else
4055         \MT@error{You cannot enable #1 if it was disabled\MessageBreak
4056             in the package options}{Load microtype with #1 enabled.}%
4057     \expandafter@gobble
4058     \fi
4059 }

4060 \MT@define@optionX{protrusion}\MT@protrudechars
4061 (/pdftex-def|luatex-def|xetex-def)
4062 (*pdftex-def|luatex-def)
4063 \MT@define@optionX{expansion}\MT@adjustspacing

```

`\MT@protrudechars`

```

\MT@adjustspacing 4064 \let\MT@protrudechars\pdfprotrudechars
4065 \let\MT@adjustspacing\pdfadjustspacing
4066 (/pdftex-def|luatex-def)
4067 (*xetex-def)
4068 \let\MT@protrudechars\XeTeXprotrudechars
4069 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
4070 (/xetex-def)

```

`\MT@define@optionX@` The same for tracking, spacing and kerning, which do not have a compatibility level.

```

4071 (*pdftex-def|luatex-def)
4072 (pdftex-def)\MT@requires@pdftex6{
4073 (luatex-def)\MT@requires@luatex3{
4074     \def\MT@define@optionX@#1#2{%
4075         \define@key{MTX}{#1}[true]{%
4076             \MT@map@clist@n{##1}{%
4077                 \KV@sp@def\MT@val{###1}%
4078                 \MT@ifempty\MT@val\relax}%
4079             \@tempcnta=\m@ne
4080             \MT@ifstreql\MT@val{true}{%
4081                 \MT@checksetup{#1}{%
4082                     \@tempcnta=\@ne
4083                     \MT@vinfo{Enabling #1\on@line}%
4084                 }%
4085             }{%
4086                 \MT@ifstreql\MT@val{false}{%
4087                     \@tempcnta=\z@
4088                     \MT@vinfo{Disabling #1\on@line}%
4089                 }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}

```

```

4090             {Use either `true' or `false'}%
4091         }%
4092     }%
4093     \ifnum\@tempcnta>\m@ne
4094     #2\relax
4095     \fi
4096 }%
4097 }%
4098 }%
4099 }

```

We cannot simply let `\MT@tracking` relax, since this may select the already letter-spaced font instance.

```

4100 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
4101         \else \let\MT@tracking\MT@tracking@ \fi}
4102 \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
4103 \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
4104         \pdfappendkern \@tempcnta}
4105 }}
4106 </pdfTeX-def|LaTeX-def>
4107 <{*pdfTeX-def|LaTeX-def|XeTeX-def}>

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

4108 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
4109 <LaTeX-def>
4110 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
4111 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
4112 <pdfTeX-def>
4113 \define@key{MTX}{activate}[true]{%
4114     \setkeys{MTX}{protrusion=#1}}%
4115 <pdfTeX-def|LaTeX-def> \setkeys{MTX}{expansion=#1}}%
4116 }
4117 </pdfTeX-def|LaTeX-def|XeTeX-def>

```

`\MT@saved@setupfont` Disable everything – may be used as a work-around in case setting up fonts doesn't work in certain environments. (*Undocumented.*)

```

4118 <{*package}>
4119 \let\MT@saved@setupfont\MT@setupfont
4120 \define@key{MTX}{disable}[]{}%
4121 \MT@info{Inactivate `~\MT@MT' package}%
4122 \let\MT@setupfont\relax
4123 }
4124 \define@key{MTX}{enable}[]{}%
4125 \MT@info{Reactivate `~\MT@MT' package}%
4126 \let\MT@setupfont\MT@saved@setupfont
4127 }
4128 </package>

```

14.4.6 Processing the options

`\MT@ProcessOptionsWithKV` Parse options.

```

4129 <{*package|Letterspace}>
4130 <plain>\MT@requires@latex1{
4131 \def\MT@ProcessOptionsWithKV#1{%
4132     \let\@tempc\relax
4133     \let\MT@temp\@empty
4134 <plain> \MT@requires@latex2{
4135     \MT@map@clist@c\@classoptionslist{%
4136         \def\CurrentOption{##1}%
4137         \MT@ifdefined@n@T{KV@#1@expandafter\MT@getkey\CurrentOption=\@nil}{%
4138             \edef\MT@temp{\MT@temp,\CurrentOption,}%
4139             \@expandtwoargs\@removeelement\CurrentOption
4140             \@unusedoptionlist\@unusedoptionlist

```

```

4141     }%
4142   }%
4143   \edef\MT@temp{\noexpand\setkeys{#1}%
4144     {\MT@temp\@ptionlist{\@currname.\@currentext}}}%

```

plain can handle package options.

```

4145 {*plain}
4146   {\edef\MT@temp{\noexpand\setkeys{#1}%
4147     {\csname usepkg@options@usepkg@pkg\endcsname}}
4148 {/plain}
4149   \MT@temp
4150   \MT@clear@options
4151 }

```

\MT@getkey For key=val in class options.

```

4152 \def\MT@getkey#1=#2\@nil{#1}
4153 \MT@ProcessOptionsWithKV{MT}
4154 {plain}\relax
4155 {/package|letterspace}
4156 {*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).

```

4157 \MT@addto@setup{%
4158 \ifMT@draft

```

We disable most of what we've just defined in the 4158 lines above if we are running in draft mode.

```

4159   \MT@warning@nl{'draft' option active.\MessageBreak
4160     Disabling all micro-typographic extensions.\MessageBreak
4161     This might lead to different line and page breaks}%
4162   \let\MT@setupfont\relax
4163   \renewcommand*\LoadMicrotypeFile[1]{}%
4164   \renewcommand*\microtypesetup[1]{}%
4165   \renewcommand*\microtypecontext[1]{}%
4166   \renewcommand*\lsstyle{}%
4167 \else
4168   \MT@setup@PDF
4169   \MT@setup@copies

```

Fix the font sets.

```

4170   \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
4171   \MT@setup@protrusion
4172   \MT@setup@expansion
4173   \MT@setup@tracking
4174   \MT@setup@wartracking
4175   \MT@setup@spacing
4176   \MT@setup@kerning
4177   \MT@setup@noligatures
4178 }
4179 {/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!

```

4180 {*pdftex-def|luatex-def}
4181 \def\MT@setup@PDF{%

```

```

4182 \MT@info@nl{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
4183           \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
4184 }

```

\MT@setup@copies Working on font copies?

```

4185 \def\MT@setup@copies{%
4186   \ifx\MT@copy@font\relax\else \MT@info@nl{Using font copies for contexts}\fi
4187 }
4188 </pdfTEX-def|LATEX-def>
4189 <*XETEX-def>
4190 \let\MT@setup@PDF\relax
4191 \let\MT@setup@copies\relax
4192 </XETEX-def>

```

\MT@setup@protrusion Protrusion.

```

4193 <*pdfTEX-def|xETEX-def|LATEX-def>
4194 \def\MT@setup@protrusion{%
4195   \ifMT@protrusion
4196     \edef\MT@active@features{\MT@active@features,pr}%
4197     \MT@protrudechars\MT@pr@level
4198     \MT@info@nl{Character protrusion enabled (level \number\MT@pr@level)%
4199               \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
4200                 factor: \number\MT@pr@factor\fi
4201               \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
4202     \MT@check@active@set{pr}%
4203   \else
4204     \let\MT@protrusion\relax
4205     \MT@info@nl{No character protrusion}%
4206   \fi
4207 }
4208 </pdfTEX-def|xETEX-def|LATEX-def>

```

\MT@setup@expansion For DVI output, the user must have explicitly passed the expansion option to the package.

```

4209 <*pdfTEX-def|LATEX-def>
4210 \def\MT@setup@expansion{%
4211   \ifnum\pdfoutput<\@ne
4212     \ifMT@opt@expansion \else
4213       \MT@expansionfalse
4214     \fi
4215   \fi
4216   \ifMT@expansion

```

Set up the values for font expansion: if stretch has not been specified, we take the default value of 20.

```

4217   \ifnum\MT@stretch=\m@ne
4218     \let\MT@stretch\MT@stretch@default
4219   \fi

```

If shrink has not been specified, it will inherit the value from stretch.

```

4220   \ifnum\MT@shrink=\m@ne
4221     \let\MT@shrink\MT@stretch
4222   \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.

```

4223   \ifnum\MT@step=\m@ne
4224 <pdfTEX-def> \MT@requires@pdfTEX6{%
4225   \def\MT@step{1 }%
4226 <*pdfTEX-def>

```

```

4227 }{%
4228   \ifnum\MT@stretch>\MT@shrink
4229     \ifnum\MT@shrink=\z@
4230       \@tempcnta=\MT@stretch
4231     \else
4232       \@tempcnta=\MT@shrink
4233     \fi
4234   \else
4235     \ifnum\MT@stretch=\z@
4236       \@tempcnta=\MT@shrink
4237     \else
4238       \@tempcnta=\MT@stretch
4239     \fi
4240   \fi
4241   \divide\@tempcnta 5\relax
4242   \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
4243   \edef\MT@step{\number\@tempcnta\space}%
4244 }%
4245 (/pdftex-def)
4246 \fi
4247 \ifnum\MT@step=\z@
4248   \MT@warning@nl{The expansion step cannot be set to zero.\MessageBreak
4249     Setting it to one}%
4250   \def\MT@step{1 }%
4251 \fi

```

`\MT@auto` Automatic expansion of the font? This new feature of pdf_T_EX 1.20 makes the *fix* programme really usable. It must be either ‘autoexpand’ or empty (or ‘1000’ for older versions of pdf_T_EX).

```

4252 \let\MT@auto\empty
4253 \ifMT@auto
4254 (pdftex-def) \MT@requires@pdftex4{%

```

We turn off automatic expansion if output mode is DVI.

```

4255   \ifnum\pdfoutput<\@ne
4256     \ifMT@opt@auto
4257       \MT@error{%
4258         Automatic font expansion only works for PDF output.\MessageBreak
4259         However, you are creating a DVI file}
4260       {If you have created expanded fonts instances, remove `auto' from%
4261         \MessageBreak the package options. Otherwise, you have to switch
4262         off expansion.\MessageBreak completely.}%
4263     \fi
4264     \MT@autofalse
4265   \else
4266     \def\MT@auto{autoexpand}%
4267   \fi

```

Also, if pdf_T_EX is too old.

```

4268 (*pdftex-def)
4269 }{%
4270   \MT@error{%
4271     The pdftex version you are using is too old for.\MessageBreak
4272     automatic font expansion}%
4273   {If you have created expanded fonts instances, remove `auto' from.\MessageBreak
4274     the package options. Otherwise, you have to switch off expansion.\MessageBreak
4275     completely, or upgrade pdftex to version 1.20 or newer.}%
4276   \MT@autofalse
4277   \def\MT@auto{1000 }%
4278 }%
4279 (/pdftex-def)
4280 \else

```

No automatic expansion.

```

4281 (*pdftex-def)

```



```

4282     \MT@requires@pdftex4\relax{%
4283       \def\MT@auto{1000 }%
4284     }%
4285 /pdftex-def
4286     \fi

```

Choose the appropriate macro for selected expansion.

```

4287     \ifMT@selected
4288       \let\MT@set@ex@codes\MT@set@ex@codes@s
4289     \else
4290       \let\MT@set@ex@codes\MT@set@ex@codes@n
4291     \fi

```

Filter out stretch=0, shrink=0, since it would result in a pdfTeX error.

```

4292     \ifnum\MT@stretch=\z@
4293       \ifnum\MT@shrink=\z@
4294         \MT@warning@n1{%
4295           Both the stretch and shrink limit are set to zero.\MessageBreak
4296           Disabling font expansion}%
4297         \MT@expansionfalse
4298       \fi
4299     \fi
4300   \fi
4301   \ifMT@expansion
4302     \edef\MT@active@features{\MT@active@features,x}%
4303     \MT@adjustspacing\MT@ex@level
4304     \MT@info@n1{\ifMT@auto A\else Non-a\fi utomatic font expansion enabled
4305               (level \number\MT@ex@level),\MessageBreak
4306               stretch: \number\MT@stretch, shrink: \number\MT@shrink,
4307               step: \number\MT@step, \ifMT@selected\else non-\fi selected}%

```

`\MT@check@step` Check whether stretch and shrink are multiples of step.

```

4308     \def\MT@check@step##1{%
4309       \@tempcnta=\csname MT@##1\endcsname
4310       \divide\@tempcnta \MT@step
4311       \multiply\@tempcnta \MT@step
4312       \ifnum\@tempcnta=\csname MT@##1\endcsname\else
4313         \MT@warning@n1{The ##1 amount is not a multiple of step.\MessageBreak
4314           The effective maximum ##1 is \the\@tempcnta\space
4315           (step \number\MT@step)}%
4316       \fi
4317     }%
4318     \MT@check@step{stretch}%
4319     \MT@check@step{shrink}%
4320     \MT@check@active@set{ex}%

```

Inside `\showhyphens`, font expansion should be disabled.

```

4321     \CheckCommand*\showhyphens[1]{\setbox0\vbox{%
4322       \color@begingroup\everypar{}\parfillskip\z@skip
4323       \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
4324       \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%

```

`\showhyphens` I wonder why it's defined globally (in `lftssbas.dtx`)?

```

4325     \gdef\showhyphens##1{\setbox0\vbox{%
4326       \color@begingroup\pdfadjustspacing\z@\everypar{}\parfillskip\z@skip
4327       \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
4328       \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
4329   \else
4330     \let\MT@expansion\relax
4331     \MT@info@n1{No font expansion}%
4332   \fi
4333 }
4334 /pdftex-def|luatex-def
4335 *xetex-def
4336 \def\MT@setup@expansion{%

```

```

4337 \ifMT@expansion
4338 \ifMT@opt@expansion
4339 \MT@error{Font expansion does not work with xetex}
4340 {Use pdftex or luatex instead.}%
4341 \fi
4342 \fi
4343 }
4344 </xetex-def>

```

`\MT@setup@tracking` Tracking, spacing and kerning.

```

4345 <*pdftex-def|luatex-def>
4346 <pdftex-def>\MT@requires@pdftex6{%
4347 <luatex-def>\MT@requires@luatex3{%
4348 \def\MT@setup@tracking{%
4349 \ifMT@tracking
4350 \edef\MT@active@features{\MT@active@features,tr}%
4351 \MT@info@nl{Tracking enabled}%
4352 \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

4353 \ifMT@protrusion\else\MT@protrudechars\@ne\fi
4354 \else
4355 \let\MT@tracking\relax
4356 \MT@info@nl{No adjustment of tracking}%
4357 \fi
4358 }
4359 </pdftex-def|luatex-def>

```

`\MT@setup@spacing`

```

4360 <*pdftex-def>
4361 \def\MT@setup@spacing{%
4362 \ifMT@spacing
4363 \edef\MT@active@features{\MT@active@features,sp}%
4364 \pdfadjustinterwordglue\@ne
4365 \MT@info@nl{Adjustment of interword spacing enabled}%

```

The ragged2e package sets interword spaces to a fixed value without glue. mi crotyle's modifications can therefore have undesired effects. Therefore, we issue a warning.

```

4366 \MT@with@package@T{ragged2e}{%
4367 \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
4368 Adjustment of interword spacing may lead to\MessageBreak
4369 undesired results when used with `ragged2e'.\MessageBreak
4370 In this case, disable the `spacing' option}%
4371 }%
4372 \MT@check@active@set{sp}%
4373 \else
4374 \let\MT@spacing\relax
4375 \MT@info@nl{No adjustment of interword spacing}%
4376 \fi
4377 }

```

`\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`.¹⁶

```

4378 \def\MT@setup@spacing@check{%
4379 \ifMT@spacing
4380 \ifMT@babel \else
4381 \ifnum\sfcode`. > 1500
4382 \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
4383 \MT@warning@nl{%
4384 \string\nonfrenchspacing\space is active. Adjustment of\MessageBreak

```

¹⁶ Cf. the c.t.t. thread '`\frenchspacing` with AMS packages and babel', started by Philipp Lehman on 16 August 2005, MID: `ddtbajrob1@online.de`

```

4385             interword spacing will disable it. You might want\MessageBreak
4386             to add ` \@backslashchar\MT@MT context{spacing=nonfrench}'\MessageBreak
4387             to your preamble)%
4388         }%
4389         \fi
4390     \fi
4391 \fi
4392 }

\MT@setup@kerning

4393 \def\MT@setup@kerning{%
4394     \ifMT@kerning
4395         \edef\MT@active@features{\MT@active@features,kn}%
4396         \pdfprependkern\@ne
4397         \pdfappendkern\@ne
4398         \MT@info@n1{Adjustment of character kerning enabled}%
4399         \MT@check@active@set{kn}%
4400     \else
4401         \let\MT@kerning\relax
4402         \MT@info@n1{No adjustment of character kerning}%
4403     \fi
4404 }
4405 \langle pdftex-def \rangle

\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.

4406 \langle pdftex-def | luatex-def \rangle {
4407     \langle *luatex-def \rangle
4408     \def\MT@setup@tracking{%
4409         \ifMT@tracking
4410             \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
4411             or newer. Switching it off}{Upgrade luatex.}%
4412             \MT@trackingfalse
4413             \MT@let@nc{MT@tracking}\relax
4414         \else
4415             \MT@info@n1{No adjustment of tracking (luatex too old)}%
4416         \fi
4417     }
4418 }
4419 \langle /luatex-def \rangle
4420 \langle *pdftex-def | xetex-def | luatex-def \rangle
4421     \def\MT@error@doesnt@work#1{%
4422         \csname ifMT@#1\endcsname
4423         \MT@error{The #1 feature only works with pdftex 1.40\MessageBreak
4424         or newer. Switching it off}
4425     \langle pdftex-def \rangle {Upgrade pdftex.}%
4426     \langle luatex-def | xetex-def \rangle {Use pdftex instead.}%
4427         \csname MT@#1false\endcsname
4428         \MT@let@nc{MT@#1}\relax
4429     \else
4430         \MT@info@n1{No adjustment of #1}
4431     \langle pdftex-def \rangle \space{pdftex too old}%
4432     }%
4433     \fi
4434 }
4435 \langle pdftex-def | xetex-def \rangle \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
4436 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
4437 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
4438 \langle pdftex-def \rangle
4439 \langle /pdftex-def | xetex-def | luatex-def \rangle

\MT@setup@warntracking

4440 \langle letterspace \rangle \MT@addto@setup
4441 \langle pdftex-def | luatex-def \rangle \def\MT@setup@warntracking

```

`\MT@warn@tracking@DVI` We issue a warning, when letterspacing in DVI mode, since it will probably not work. We also switch on protrusion if it isn't already, to compensate for the letterspacing kerns.

```

4442 <*pdftex-def|luatex-def|letterspace>
4443 {%
4444   \ifnum\pdfoutput<\@ne
4445     \def\MT@warn@tracking@DVI{%
4446       \MT@warning@n1{%
4447         You are using tracking/letterspacing in DVI mode.\MessageBreak
4448         This will probably not work, unless the post-\MessageBreak
4449         processing program (dvips, dviptfm(x), ...) is\MessageBreak
4450         able to create the virtual fonts on the fly}%
4451       \MT@gl@et\MT@warn@tracking@DVI\relax
4452     }%
4453   \else
4454     \def\MT@warn@tracking@DVI{%
4455       \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
4456       \MT@gl@et\MT@warn@tracking@DVI\relax
4457     }%
4458   \fi
4459   \ifnum\MT@letterspace=\m@ne
4460     \let\MT@letterspace\MT@letterspace@default
4461   \else
4462     \MT@ls@too@large\MT@letterspace
4463   \fi
4464 }
4465 </pdftex-def|luatex-def|letterspace>
4466 <xetex-def>\let\MT@setup@warn@tracking\relax

```

`\MT@setup@noligatures` `\DisableLigatures` is only admissible in the preamble, therefore we can now disable the corresponding macro, if it was never called.

```

4467 <*pdftex-def|luatex-def>
4468 \def\MT@setup@noligatures{%
4469 <pdftex-def> \MT@requires@pdftex5{%
4470   \if\MT@noligatures \else
4471     \let\MT@noligatures\relax
4472   \fi
4473 <pdftex-def> }\relax
4474 }
4475 </pdftex-def|luatex-def>
4476 <xetex-def>\let\MT@setup@noligatures\relax

```

Remove the leading comma in `\MT@active@features`, and set the document switch to true.

```

4477 <*package>
4478 \MT@addto@setup{%
4479   \ifx\MT@active@features\@empty \else
4480     \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
4481   \fi
4482   \MT@documenttrue
4483 }

```

`\MT@set@babel@context` Interaction with babel.

```

4484 \def\MT@set@babel@context#1{%
4485   \MT@ifdefined@n@TF{MT@babel@#1}{%
4486     \MT@vinfo{*** Changing to language context `#1'\MessageBreak\on@line}%
4487     \expandafter\MT@exp@one@n\expandafter\microtypecontext
4488     \csname MT@babel@#1\endcsname
4489   }{%
4490     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
4491   }%
4492 }

```

`\MT@shorthandoff` Active characters can only be switched off if babel isn't loaded after microtype.

```

4493 \ifpackageloaded{babel}{
4494   \def\MT@shorthandoff#1#2{%
4495     \MT@info@n1{Switching off #1 babel's active characters (#2)}%
4496     \shorthandoff{#2}}
4497 }{
4498   \def\MT@shorthandoff#1#2{%
4499     \MT@error{You must load `babel' before `~\MT@MT'}
4500     {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
4501       active characters.}}
4502 }

```

We patch the language switching commands to enable language-dependent setup.

```

4503 \MT@addto@setup{%
4504   \ifMT@babel
4505     \ifpackageloaded{babel}{%
4506       \MT@info@n1{Redefining babel's language switching commands}%
4507       \let\MT@orig@select@language\select@language
4508       \def\select@language#1{%
4509         \MT@orig@select@language{#1}%
4510         \MT@set@babel@context{#1}%
4511       }%
4512       \let\MT@orig@foreign@language\foreign@language
4513       \def\foreign@language#1{%
4514         \MT@orig@foreign@language{#1}%
4515         \MT@set@babel@context{#1}%
4516       }%
4517     \ifMT@kerning

```

Disable French babel's active characters.

```

4518     \MT@if@false
4519     \MT@with@babel@and@T{french} \MT@if@true
4520     \MT@with@babel@and@T{frenchb} \MT@if@true
4521     \MT@with@babel@and@T{français}\MT@if@true
4522     \MT@with@babel@and@T{canadien}\MT@if@true
4523     \MT@with@babel@and@T{acadian} \MT@if@true
4524     \ifMT@if@~\MT@shorthandoff{French}{:;!?}\fi

```

Disable Turkish babel's active characters.

```

4525     \MT@if@false
4526     \MT@with@babel@and@T{turkish} \MT@if@true
4527     \ifMT@if@~\MT@shorthandoff{Turkish}{:!=}\fi
4528     \fi

```

In case babel was loaded before microtype:

```

4529     \MT@set@babel@context\languagename
4530   }{%
4531     \MT@warning@n1{You did not load the babel package.\MessageBreak
4532       The `babel' option won't have any effect}%
4533   }%
4534   \fi
4535 }

```

Now we close the `\fi` from `\ifMT@draft`.

```

4536 \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.

```

4537   \selectfont}

```

`\MT@curr@file` This is the current file (hopefully with the correct extension).

```

4538 \edef\MT@curr@file{\jobname.tex}
4539 </package>

```

Finally, execute the setup macro at the end of the preamble, and empty it (the combine class calls it repeatedly).

```
4540 <*package|letterspace>
4541 <plain>\MT@requires@l@texl{
4542 \AtBeginDocument{\MT@setup@ \MT@gl@et\MT@setup@\@empty}
4543 <plain>}\relax
4544 </package|letterspace>
```

Must come at the very, very end.

```
4545 <package>\MT@ifdefined@c@T\MT@setup@spacing@check
4546 <package> { \AtBeginDocument{\MT@setup@spacing@check} }
```

Restore catcodes.

```
4547 <package|letterspace>\MT@restore@catcodes
```

That was that.

15 Configuration files

Let's now write the font configuration files.

```
4548 <*config>
4549
```

15.1 Font sets

We first declare some sets in the main configuration file.

```
4550 <*m-t>
4551 %%% -----
4552 %%% FONT SETS
4553
4554 \DeclareMicrotypeSet{all}
4555 { }
4556
4557 \DeclareMicrotypeSet{allmath}
4558 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TS1,OML,OMS,U} }
4559
4560 \DeclareMicrotypeSet{alltext}
4561 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2} }
4562
4563 \DeclareMicrotypeSet{basicmath}
4564 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,OML,OMS},
4565   family = {rm*,sf*},
4566   series = {md*},
4567   size = {normalsize,footnotesize,small,large}
4568 }
4569
4570 \DeclareMicrotypeSet{basictext}
4571 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2},
4572   family = {rm*,sf*},
4573   series = {md*},
4574   size = {normalsize,footnotesize,small,large}
4575 }
4576
4577 \DeclareMicrotypeSet{smallcaps}
4578 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2},
4579   shape = {sc*}
4580 }
4581
```

```

4582 \DeclareMicrotypeSet{footnotesize}
4583   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2},
4584     size      = {-small}
4585   }
4586
4587 \DeclareMicrotypeSet{scriptsize}
4588   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2},
4589     size      = {-footnotesize}
4590   }
4591
4592 \DeclareMicrotypeSet{normal font}
4593   { font = */*/*/*/* }
4594

```

The default sets.

```

4595 %%% -----
4596 %%% DEFAULT SETS
4597
4598 \DeclareMicrotypeSetDefault[protrusion]{alltext}
4599 \DeclareMicrotypeSetDefault[expansion]{basictext}
4600 \DeclareMicrotypeSetDefault[spacing]{basictext}
4601 \DeclareMicrotypeSetDefault[kerning]{alltext}
4602 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
4603

```

15.2 Font variants and aliases

```

4604 %%% -----
4605 %%% FONT VARIANTS AND ALIASES
4606

```

These are the variants I happen to be using (expert encoding, oldstyle numerals, swashes, alternative, display, inferior and superior numerals):

```

4607 \DeclareMicrotypeVariants{x,j,w,a,d,0,1}
4608

```

Other candidates: 2 (proportional digits), e (engraved), f (Fraktur), g (small text), h (shadow), l (outline), n (informal), p (ornaments), r (roman), s (sans serif), t (typewriter). I've omitted them since they seem hardly be used and/or they are actually more than just a variant, i.e., they shouldn't share a file.

Fonts that are 'the same': The fontspec package will use lmr by default, whose EU1/2 encoding is declared in mt-LatinModernRoman.cfg.

```

4609 \ifMT@fontspec
4610 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
4611 \else
4612 \DeclareMicrotypeAlias{lmr}{cmr} % lmodern
4613 \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.

```

4614 \DeclareMicrotypeAlias{lmsy}{cmsy}
4615 \DeclareMicrotypeAlias{lmm}{cmm}
4616 \DeclareMicrotypeAlias{aer}{cmr} % ae
4617 \DeclareMicrotypeAlias{zer}{cmr} % zefonts
4618 \DeclareMicrotypeAlias{cmor}{cmr} % eco
4619 \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).

```
4620 \DeclareMicrotypeAlias{pxr} {ppl} % pxfonts
4621 \DeclareMicrotypeAlias{qpl} {ppl} % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)
```

The ‘FPL Neu’ fonts, a ‘re-implementation’ of Palatino.

```
4622 \DeclareMicrotypeAlias{fp9x}{pplx} % FPL Neu
4623 \DeclareMicrotypeAlias{fp9j}{pplj} % "
4624 \DeclareMicrotypeAlias{txr} {ptm} % txfonts
4625 \DeclareMicrotypeAlias{qtm} {ptm} % TeX Gyre Termes (formerly: qfonts/QuasiTimes)
```

The OpenType versions:

```
4626 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino Linotype}
4627 \DeclareMicrotypeAlias{Palatino LT Std} {Palatino Linotype}
4628 \DeclareMicrotypeAlias{Palatino}      {Palatino Linotype}
4629 \DeclareMicrotypeAlias{Asana Math}    {Palatino Linotype}
```

More Times variants, to be checked: pns, mns (TimesNewRomanPS); mnt (TimesNewRomanMT, TimesNRSevenMT), mtm (TimesSmallTextMT); pte (TimesEuropa); ptt (TimesTen); TimesEighteen; TimesModernEF.

The `eulervm` package virtually extends the Euler fonts.

```
4630 \DeclareMicrotypeAlias{zeur}{eur} % Euler VM
4631 \DeclareMicrotypeAlias{zeus}{eus} % "
```

MicroPress’s Charter version (chmath).

```
4632 \DeclareMicrotypeAlias{chr} {bch} % CH Math
```

The `mathdesign` package provides math fonts matching Bitstream Charter and URW Garamond.

```
4633 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
4634 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond
```

The `garamondx` package, an extension of URW Garamond, providing small caps and oldstyle figures.

```
4635 \DeclareMicrotypeAlias{zgmX}{ugm} % garamondx
4636 \DeclareMicrotypeAlias{zgmj}{ugm} % "
4637 \DeclareMicrotypeAlias{zgmI}{ugm} % "
4638 \DeclareMicrotypeAlias{zgmQ}{ugm} % "
```

URW Letter Gothic is similar enough to Bitstream Letter Gothic to share the configuration.

```
4639 \DeclareMicrotypeAlias{ulg} {blg} % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

Euro symbol fonts, to save some files.

```
4640 \DeclareMicrotypeAlias{zpeu} {zpeu} % Adobe Euro sans -> serif
4641 \DeclareMicrotypeAlias{eurosans}{zpeu} % Adobe Euro sans -> serif
4642 \DeclareMicrotypeAlias{euroitcs}{euroitc} % ITC Euro sans -> serif
4643
```

15.3 Interaction with babel

Contexts that are to be set when switching to a language.

```
4644 %%% -----
4645 %%% INTERACTION WITH THE `babel' PACKAGE
4646
4647 \DeclareMicrotypeBabelHook
4648   {english,UKenglish,british,USenglish,american}
4649   {kerning=, spacing=nonfrench}
4650
4651 \DeclareMicrotypeBabelHook
4652   {french,français,acadian,canadien}
4653   {kerning=french, spacing=}
4654
4655 \DeclareMicrotypeBabelHook
4656   {turkish}
4657   {kerning=turkish, spacing=}
4658
```


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`d symbols have to be specified numerically. Of course, numerical identification is possible in any other case, too.

8-bit characters are also admissible, provided they have been declared in the input encoding file. They should, however, only be used in private configuration files, where the proper input encoding is guaranteed, or else in combination with the 'inputenc' key.

15.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not \mathbb{C} for \mathbb{O} .

```
4659 </m-t>
4660 <*/m-t|zpeu|mvs>
4661 %%% -----
4662 %%% CHARACTER INHERITANCE
4663
4664 </m-t|zpeu|mvs>
4665 <*/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'), \mathbb{A} , \mathbb{a} , \mathbb{C} , \mathbb{c} .

```
4666 \DeclareCharacterInheritance
4667 { encoding = OT1 }
4668 { f = {011}, % ff
4669   i = {\i},
4670   j = {\j},
4671   0 = {\0},
4672   o = {\o}
4673 }
4674
```

15.5.2 T1

Candidates here: 028 ('fi'), 029 ('fi'), 030 ('ffi'), 031 ('ffl'), 156 ('IJ' ligature, since \LaTeX 2005/12/01 accessible as \IJ), 188 ('ij', \i j), \mathcal{A} , \mathcal{a} , \mathcal{C} , \mathcal{c} .

```

4675 \DeclareCharacterInheritance
4676   { encoding = T1 }
4677   { A = {\^A,\'A,\^A,\-A,\"A,\r A,\k A,\u A},
4678     a = {\`a,\'a,\^a,\-a,\"a,\r a,\k a,\u a},
4679     C = {\'C,\c C,\v C},
4680     c = {\'c,\c c,\v c},
4681     D = {\v D,\DH},
4682     d = {\v d,\dj},
4683     E = {\^E,\'E,\^E,\"E,\k E,\v E},
4684     e = {\`e,\'e,\^e,\"e,\k e,\v e},
4685     f = {027}, % ff
4686     G = {\u G},
4687     g = {\u g},
4688     I = {\^I,\'I,\^I,\"I,\.I},
4689     i = {\`i,\'i,\^i,\"i,\i},
4690     j = {\j},
4691     L = {\L,\'L,\v L},
4692     l = {\l,\'l,\v l},
4693     N = {\'N,\-N,\v N},
4694     n = {\'n,\-n,\v n},
4695     O = {\0,\^0,\'0,\^0,\-0,\"0,\H 0},
4696     o = {\0,\`o,\'o,\^o,\-o,\"o,\H o},
4697     R = {\'R,\v R},
4698     r = {\'r,\v r},
4699     S = {\'S,\c S,\v S,\SS},
4700     s = {\'s,\c s,\v s},
4701     T = {\c T,\v T},
4702     t = {\c t,\v t},
4703     U = {\^U,\'U,\^U,\"U,\H U,\r U},
4704     u = {\`u,\'u,\^u,\"u,\H u,\r u},
4705     Y = {\'Y,\"Y},
4706     y = {\'y,\"y},
4707     Z = {\'Z,\.Z,\v Z},
4708     z = {\'z,\.z,\v z}

```

The ‘soft hyphen’ often has reduced right side bearing so that it may already be protruded, hence no inheritance.

```

4709 %   - = {127},
4710   }
4711

```

15.5.3 LY1

More characters: 008 ('fl'), 012 ('fi'), 014 ('ffi'), 015 ('ffl'), \mathcal{A} , \mathcal{a} , \mathcal{C} , \mathcal{c} .

```

4712 \DeclareCharacterInheritance
4713   { encoding = LY1 }
4714   { A = {\^A,\'A,\^A,\-A,\"A,\r A},
4715     a = {\`a,\'a,\^a,\-a,\"a,\r a},
4716     C = {\c C},
4717     c = {\c c},
4718     D = {\DH},
4719     E = {\^E,\'E,\^E,\"E},
4720     e = {\`e,\'e,\^e,\"e},
4721     f = {011}, % ff
4722     I = {\^I,\'I,\^I,\"I},
4723     i = {\`i,\'i,\^i,\"i,\i},
4724     L = {\L},
4725     l = {\l},
4726     N = {\-N},

```

```

4727     n = {\-n},
4728     O = {\^0,\'0,\^0,\-0,\"0,\0},
4729     o = {\^o,\'o,\^o,\-o,\"o,\o},
4730     S = {\v S},
4731     s = {\v s},
4732     U = {\^U,\'U,\^U,\"U},
4733     u = {\^u,\'u,\^u,\"u},
4734     Y = {\'Y,\"Y},
4735     y = {\'y,\"y},
4736     Z = {\v Z},
4737     z = {\v z}
4738   }
4739

```

15.5.4 OT4

The Polish OT1 extension. More interesting characters here: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Ć, ć.

```

4740 \DeclareCharacterInheritance
4741   { encoding = OT4 }
4742   { A = {\k A},
4743     a = {\k a},
4744     C = {\'C},
4745     c = {\'c},
4746     E = {\k E},
4747     e = {\k e},
4748     f = {011}, % ff
4749     i = {\i},
4750     j = {\j},
4751     L = {\L},
4752     l = {\l},
4753     N = {\'N},
4754     n = {\'n},
4755     O = {\0,\"0},
4756     o = {\o,\"o},
4757     S = {\'S},
4758     s = {\'s},
4759     Z = {\'Z,\"Z},
4760     z = {\'z,\"z}
4761   }
4762

```

15.5.5 QX

The Central European QX encoding.¹⁷ Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Ć, ć.

```

4763 \DeclareCharacterInheritance
4764   { encoding = QX }
4765   { A = {\^A,\"A,\^A,\-A,\"A,\k A,\"AA},
4766     a = {\^a,\"a,\^a,\-a,\"a,\k a,\"aa},
4767     C = {\'C,\"C},
4768     c = {\'c,\"c},
4769     D = {\DH},
4770     E = {\^E,\"E,\^E,\"E,\k E},
4771     e = {\^e,\"e,\^e,\"e,\k e},
4772     f = {011}, % ff
4773     I = {\^I,\"I,\^I,\"I,\k I},
4774     i = {\^i,\"i,\^i,\"i,\k i,\"i},
4775     j = {\j},
4776     L = {\L},

```

17 Contributed by Maciej Eder.

```

4777 l = {\l},
4778 N = {\'N,\-N},
4779 n = {\'n,\-n},
4780 O = {\0,\^0,\'0,\^0,\-0,\"0},
4781 o = {\o,\^o,\'o,\^o,\-o,\"o},

```

The Rumanian `\textcommabelow` accents are actually replacements for the `\c` variants, which had previously (and erroneously¹⁸) been included in QX encoding. They are still kept for backwards compatibility.

```

4782 S = {\'S,\c S,\textcommabelow S,\v S},
4783 s = {\'s,\c s,\textcommabelow s,\v s},
4784 T = {\c T,\textcommabelow T},
4785 t = {\c t,\textcommabelow t},
4786 U = {\^U,\'U,\^U,\"U,\k U},
4787 u = {\^u,\'u,\^u,\"u,\k u},
4788 Y = {\'Y,\"Y},
4789 y = {\'y,\"y},
4790 Z = {\'Z,\.Z,\v Z},
4791 z = {\'z,\.z,\v z},
4792 . = \textellipsis
4793 }
4794

```

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.

```

4795 \DeclareCharacterInheritance
4796 { encoding = T5 }
4797 { A = {\^A,\'A,\-A,\h A,\d A,\^A,\u A,
4798       \^Acircumflex,\'Acircumflex,\-Acircumflex,\hAcircumflex,\dAcircumflex,
4799       \^Abreve,\'Abreve,\-Abreve,\hAbreve,\dAbreve},
4800 a = {\^a,\'a,\-a,\h a,\d a,\^a,\u a,
4801       \^acircumflex,\'acircumflex,\-acircumflex,\hacircumflex,\dacircumflex,
4802       \^abreve,\'abreve,\-abreve,\h\abreve,\d\abreve},
4803 D = {\DJ},
4804 d = {\dj},
4805 E = {\^E,\'E,\-E,\h E,\d E,\^E,
4806       \^Ecircumflex,\'Ecircumflex,\-Ecircumflex,\hEcircumflex,\dEcircumflex},
4807 e = {\^e,\'e,\-e,\h e,\d e,\^e,
4808       \^ecircumflex,\'ecircumflex,\-ecircumflex,\h\ecircumflex,\d\ecircumflex},
4809 I = {\^I,\'I,\-I,\h I,\d I},
4810 i = {\^i,\'i,\-i,\h i,\d i,\i},
4811 O = {\^O,\'O,\-O,\h O,\d O,\^O,\horn O,
4812       \^Ocircumflex,\'Ocircumflex,\-Ocircumflex,\hOcircumflex,\dOcircumflex,
4813       \^Ohorn,\'Ohorn,\-Ohorn,\hOhorn,\dOhorn},
4814 o = {\^o,\'o,\-o,\h o,\d o,\^o,\horn o,
4815       \^ocircumflex,\'ocircumflex,\-ocircumflex,\h\ocircumflex,\d\ocircumflex,
4816       \^ohorn,\'ohorn,\-ohorn,\h\ohorn,\d\ohorn},
4817 U = {\^U,\'U,\-U,\h U,\d U,\horn U,
4818       \^Uhorn,\'Uhorn,\-Uhorn,\hUhorn,\dUhorn},
4819 u = {\^u,\'u,\-u,\h u,\d u,\horn u,
4820       \^uhorn,\'uhorn,\-uhorn,\h\uhorn,\d\uhorn},
4821 Y = {\^Y,\'Y,\-Y,\h Y,\d Y},
4822 y = {\^y,\'y,\-y,\h y,\d y}
4823 }
4824

```

18 Cf. <http://tug.org/pipermail/tex-live/2008-August/017204.html>

15.5.7 EU1, EU2

The EU1 and EU2 encodings are not well-defined as they don't contain a fixed number of glyphs, all of which must be present. OpenType fonts may contain thousands of glyphs, but we only define those that should be present in every font (basically T1). This inheritance list should be overridden by font-specific ones.

```

4825 \DeclareCharacterInheritance
4826   { encoding = {EU1,EU2} }
4827   { A = {\^A,\A,\^A,\^A,\-A,\"A,\r A,\k A,\u A},
4828     a = {\^a,\'a,\^a,\-a,\"a,\r a,\k a,\u a},
4829     C = {\'C,\c C,\v C},
4830     c = {\'c,\c c,\v c},
4831     D = {\v D,\DH},
4832     d = {\v d,\dj},
4833     E = {\^E,\'E,\^E,\^E,\k E,\v E},
4834     e = {\^e,\'e,\^e,\^e,\k e,\v e},
4835 %   f = {/f_f}, % sometimes /f_f, sometimes /ff
4836     G = {\u G},
4837     g = {\u g},
4838     I = {\^I,\'I,\^I,\^I,\"I,\.I},
4839     i = {\^i,\'i,\^i,\^i,\"i,\i},
4840 %   j = {\j},
4841     L = {\L,\'L,\v L},
4842     l = {\l,\'l,\v l},
4843     N = {\'N,\-N,\v N},
4844     n = {\'n,\-n,\v n},
4845     O = {\0,\^0,\'0,\^0,\-0,\"0,\H 0},
4846     o = {\0,\^o,\'o,\^o,\-o,\"o,\H o},
4847     R = {\'R,\v R},
4848     r = {\'r,\v r},
4849     S = {\'S,\c S,\v S}, % \SS
4850     s = {\'s,\c s,\v s},
4851     T = {\c T,\v T},
4852     t = {\c t,\v t},
4853     U = {\^U,\'U,\^U,\^U,\"U,\H U,\r U},
4854     u = {\^u,\'u,\^u,\^u,\"u,\H u,\r u},
4855     Y = {\'Y,\"Y},
4856     y = {\'y,\"y},
4857     Z = {\'Z,\.Z,\v Z},
4858     z = {\'z,\.z,\v z}
4859   }
4860
4861 </m-t>

```

15.5.8 Euro symbols

Make Euro symbols settings simpler.

```

4862 < *zpeu >
4863 \DeclareCharacterInheritance
4864   { encoding = U,
4865     family = {zpeu,zpeus,eurosans} }
4866   { E = 128 }
4867
4868 < /zpeu >
4869 < *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.

```

4870 \DeclareCharacterInheritance
4871   { encoding = {OT1,U},
4872     family = mvs }

```

```

4873 { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
4874
4875 (/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.

```

4876 (*m-t)
4877 %%% -----
4878 %%% TRACKING/LETTERSPACING
4879
4880 \SetTracking
4881 [ name = default,
4882   no ligatures = {f} ]
4883 { encoding = {OT1,T1,T2A,LY1,OT4,QX,EU2} }
4884 { }
4885

```

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).

```

4886 %%% -----
4887 %%% EXPANSION
4888
4889 \SetExpansion
4890 [ name = default ]
4891 { encoding = {OT1,OT4,QX,T1,LY1} }
4892 {
4893   A = 500,   a = 700,
4894   \AE = 500, \ae = 700,
4895   B = 700,   b = 700,
4896   C = 700,   c = 700,
4897   D = 500,   d = 700,
4898   E = 700,   e = 700,
4899   F = 700,
4900   G = 500,   g = 700,
4901   H = 700,   h = 700,
4902   K = 700,   k = 700,
4903   M = 700,   m = 700,
4904   N = 700,   n = 700,
4905   O = 500,   o = 700,
4906   \OE = 500, \oe = 700,
4907   P = 700,   p = 700,
4908   Q = 500,   q = 700,
4909   R = 700,
4910   S = 700,   s = 700,
4911   U = 700,   u = 700,
4912   W = 700,   w = 700,
4913   Z = 700,   z = 700,
4914   2 = 700,
4915   3 = 700,
4916   6 = 700,
4917   8 = 700,
4918   9 = 700
4919 }
4920

```

Settings for Cyrillic T2A encoding.¹⁹

19 Contributed by *Karl Karlsson*.

```

4921 \SetExpansion
4922   [ name      = T2A ]
4923   { encoding = T2A }
4924   {
4925     A = 500,      a = 700,
4926     B = 700,      b = 700,
4927     C = 700,      c = 700,
4928     D = 500,      d = 700,
4929     E = 700,      e = 700,
4930     F = 700,
4931     G = 500,      g = 700,
4932     H = 700,      h = 700,
4933     K = 700,      k = 700,
4934     M = 700,      m = 700,
4935     N = 700,      n = 700,
4936     O = 500,      o = 700,
4937     P = 700,      p = 700,
4938     Q = 500,      q = 700,
4939     R = 700,
4940     S = 700,      s = 700,
4941     U = 700,      u = 700,
4942     W = 700,      w = 700,
4943     Z = 700,      z = 700,
4944     2 = 700,
4945     3 = 700,
4946     6 = 700,
4947     8 = 700,
4948     9 = 700,
4949     \CYRA = 500,   \cyra = 700,
4950     \CYRB = 700,   \cyrb = 700,
4951     \CYRV = 700,   \cyrv = 700,
4952     \CYRG = 700,   \cyrg = 700,
4953     \CYRD = 700,   \cyrd = 700,
4954     \CYRE = 700,   \cyre = 700,
4955     \CYRZH = 700,  \cyrzh = 700,
4956     \CYRZ = 700,   \cyrz = 700,
4957     \CYRI = 700,   \cyri = 700,
4958     \CYRISHRT = 700, \cyrishrt = 700,
4959     \CYRK = 700,   \cyrk = 700,
4960     \CYRL = 700,   \cyr| = 700,
4961     \CYRM = 700,   \cyrm = 700,
4962     \CYRN = 700,   \cyrn = 700,
4963     \CYRO = 500,   \cyro = 700,
4964     \CYRP = 700,   \cyrp = 700,
4965     \CYRR = 700,   \cyr| = 700,
4966     \CYRS = 700,   \cyr| = 700,
4967     \CYRT = 700,   \cyrt = 700,
4968     \CYRU = 700,   \cyru = 700,
4969     \CYRF = 700,   \cyrf = 700,
4970     \CYRH = 700,   \cyrh = 700,
4971     \CYRC = 700,   \cyrc = 700,
4972     \CYRCH = 700,  \cyrch = 700,
4973     \CYRSH = 700,  \cyrsh = 700,
4974     \CYRSHCH = 700, \cyrshch = 700,
4975     \CYRHRDSN = 700, \cyrhrdsn = 700,
4976     \CYRERY = 700,  \cyrery = 700,
4977     \CYRSFTSN = 700, \cyr| = 700,
4978     \CYREREV = 700, \cyrerev = 700,
4979     \CYRYU = 700,   \cyryu = 700,
4980     \CYRYA = 700,   \cyrya = 700
4981   }
4982

```

T5 encoding does not contain \AE, \ae, \OE and \oe.

```

4983 \SetExpansion

```

```

4984 [ name      = T5 ]
4985 { encoding = T5 }
4986 {
4987   A = 500,      a = 700,
4988   B = 700,      b = 700,
4989   C = 700,      c = 700,
4990   D = 500,      d = 700,
4991   E = 700,      e = 700,
4992   F = 700,
4993   G = 500,      g = 700,
4994   H = 700,      h = 700,
4995   K = 700,      k = 700,
4996   M = 700,      m = 700,
4997   N = 700,      n = 700,
4998   O = 500,      o = 700,
4999   P = 700,      p = 700,
5000   Q = 500,      q = 700,
5001   R = 700,
5002   S = 700,      s = 700,
5003   U = 700,      u = 700,
5004   W = 700,      w = 700,
5005   Z = 700,      z = 700,
5006   2 = 700,
5007   3 = 700,
5008   6 = 700,
5009   8 = 700,
5010   9 = 700
5011 }
5012
5013 </m-t>

```

15.8 Character protrusion

```

5014 %%% -----
5015 %%% PROTRUSION
5016

```

For future historians, Hàn Thê 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.

```

5017 <*cfg-t>
5018 \SetProtrusion
5019 <m-t> [ name = default ]

```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```

5020 <bch> [ name = bch-default ]

```

- Bitstream Letter Gothic (blg)

```

5021 <blg> [ name = blg-default ]

```

- Computer Modern Roman (cmr)

```

5022 <cmr> [ name = cmr-default ]

```

- Adobe Garamond (pad, padx, padj)

```

5023 <pad> [ name = pad-default ]

```

- Minion²⁰ (pmnx, pmnj)

```

5024 <pmn> [ name = pmnj-default ]

```

- Palatino (ppl, pplx, pplj)

```

5025 <ppl> [ name = ppl-default ]

```

- Times (ptm, ptmx, ptmj)

```

5026 <ptm> [ name = ptm-default ]

```

- URW Garamond (ugm)

```

5027 <ugm> [ name = ugm-default ]
5028 <m-t|cmr|pmn> { }
5029 <bch|blg|pad|ugm> { encoding = OT1,
5030 <ppl|ptm> { encoding = {OT1,OT4},
5031 <bch> family = bch }
5032 <blg> family = blg }
5033 <pad> family = {pad,padx,padj }
5034 <ppl> family = {ppl,pplx,pplj }
5035 <ptm> family = {ptm,ptmx,ptmj }
5036 <ugm> family = ugm }
5037 {
5038 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm> A = {50,50},
5039 <ugm> A = {50,100},
5040 <pad|ptm> \AE = {50, },
5041 <ugm> \AE = {150,50},
5042 <ugm> B = { ,50},
5043 <bch|pad|pmn|ugm> C = {50, },
5044 <bch|pad|pmn> D = { ,50},
5045 <ugm> D = { ,70},
5046 <ugm> E = { ,50},

```

```

5047 <m-t|bch|cmr|pad|pmn|ptm>    F = { ,50},
5048 <ugm>          F = { ,70},
5049 <bch|pad|pmn>    G = {50, },
5050 <ugm>          G = {50,50},
5051 <blg>          I = {150,150},
5052 <m-t|cmr|pad|pmn|ppl|ptm|ugm>    J = {50, },
5053 <bch|blg>      J = {100, },
5054 <!blg>        K = { ,50},
5055 <blg>          K = {50, },
5056 <m-t|bch|cmr|pad|pmn|ppl>    L = { ,50},
5057 <blg>          L = { ,150},
5058 <ptm>          L = { ,80},
5059 <ugm>          L = { ,120},
5060 <bch|pad|pmn|ugm>    O = {50,50},
5061 <pad>         \OE = {50, },
5062 <ugm>         \OE = {50,50},
5063 <blg>          P = { ,100},
5064 <ugm>          P = { ,50},
5065 <bch|pad|pmn>    Q = {50,70},
5066 <ugm>          Q = {50,50},
5067 <bch>          R = { ,50},
5068 <ugm>          R = { ,70},
5069 <m-t|bch|cmr|pad|pmn|ppl|ptm>    T = {50,50},
5070 <blg>          T = {100,100},
5071 <ugm>          T = {70,70},
5072 <m-t|bch|cmr|pad|pmn|ppl|ptm>    V = {50,50},
5073 <blg|ugm>     V = {70,70},
5074 <m-t|bch|cmr|pad|pmn|ppl|ptm>    W = {50,50},
5075 <ugm>          W = {70,70},
5076 <m-t|bch|cmr|pad|pmn|ppl|ptm>    X = {50,50},
5077 <ugm>          X = {50,70},
5078 <m-t|bch|cmr|pad|pmn|ppl>      Y = {50,50},
5079 <blg|ptm|ugm>   Y = {80,80},
5080 <ugm>          Z = {50,50},
5081 <blg>          f = {150,100},
5082 <blg>          i = {150,150},
5083 <blg>          j = {100,100},
5084 <m-t|bch|cmr|pad|pmn|ppl|ptm>    k = { ,50},
5085 <ugm>          k = { ,70},
5086 <blg>          l = {150,150},
5087 <pmn>          l = { , -50},
5088 <pad|ppl>      p = {50,50},
5089 <ugm>          p = { ,50},
5090 <pad|ppl>      q = {50, },
5091 <!blg>        r = { ,50},
5092 <blg>          r = {100, 80},
5093 <cmr|pad|pmn>   t = { ,70},
5094 <bch>          t = { ,50},
5095 <blg>          t = {150, 80},
5096 <ugm>          t = { ,100},
5097 <m-t|bch|cmr|pad|pmn|ppl|ptm>    v = {50,50},
5098 <blg>          v = {100,100},
5099 <ugm>          v = {50,70},
5100 <m-t|bch|cmr|pad|pmn|ppl|ptm>    w = {50,50},
5101 <ugm>          w = {50,70},
5102 <!blg>        x = {50,50},
5103 <blg>          x = {100,100},
5104 <m-t|bch|pad|pmn>   y = { ,50},
5105 <blg>          y = { 50,100},
5106 <cmr|ppl|ptm>    y = {50,70},
5107 <ugm>          y = { ,70},

5108 <cmr>          0 = { ,50},
5109 <m-t>          1 = {50,50},
5110 <bch|blg|pad|ptm|ugm>    1 = {150,150},
5111 <cmr>          1 = {100,200},

```

```

5112 <pmn>      1 = { ,50},
5113 <ppl>       1 = {100,100},
5114 <bch|cmr|pad|ugm> 2 = {50,50},
5115 <blg>      2 = { ,100},
5116 <bch|pmn>   3 = {50, },
5117 <cmr|pad|ugm> 3 = {50,50},
5118 <blg>      3 = {100, },
5119 <m-t|pad>   4 = {50,50},
5120 <bch>       4 = {100,50},
5121 <blg>      4 = {100, },
5122 <cmr|ugm>  4 = {70,70},
5123 <pmn>      4 = {50, },
5124 <ptm>      4 = {70, },
5125 <cmr>      5 = { ,50},
5126 <pad>      5 = {50,50},
5127 <bch>      6 = {50, },
5128 <cmr>      6 = { ,50},
5129 <pad>      6 = {50,50},
5130 <m-t>      7 = {50,50},
5131 <bch|pad|pmn|ugm> 7 = {50,80},
5132 <blg>      7 = {100,100},
5133 <cmr|ptm>  7 = {50,100},
5134 <ppl>      7 = { ,50},
5135 <cmr>      8 = { ,50},
5136 <bch|pad>  9 = {50,50},
5137 <cmr>     10 = { ,50},
5138 <m-t|cmr|pad|pmn|ppl|ptm|ugm> . = { ,700},
5139 <bch>      . = { ,600},
5140 <blg>     . = {400,500},
5141 <!blg>    {,}= { ,500},
5142 <blg>    {,}= {300,400},
5143 <m-t|cmr|pad|pmn|ppl|ptm|ugm> : = { ,500},
5144 <bch>     : = { ,400},
5145 <blg>     : = {300,400},
5146 <m-t|bch|pad|pmn|ptm> ; = { ,300},
5147 <blg>    ; = {200,300},
5148 <cmr|ppl> ; = { ,500},
5149 <ugm>     ; = { ,400},
5150 <!blg>   ! = { ,100},
5151 <blg>    ! = {200,200},
5152 <m-t|pad|pmn|ptm> ? = { ,100},
5153 <bch|cmr|ppl|ugm> ? = { ,200},
5154 <blg>    ? = {150,150},
5155 <pmn>    " = {300,300},
5156 <m-t|bch|cmr|pad|pmn|ppl> @ = {50,50},
5157 <ptm>   @ = {100,100},
5158 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm> ~ = {200,250},
5159 <ugm>    ~ = {300,350},
5160 <pad|ppl|ptm> & = {50,100},
5161 <ugm>    & = { ,100},
5162 <m-t|cmr|pad|pmn> \% = {50,50},
5163 <bch>    \% = { ,50},
5164 <ppl|ptm> \% = {100,100},
5165 <ugm>    \% = {50,100},
5166 <blg>    \% = {100,100},
5167 <m-t|ppl|ptm|ugm> * = {200,200},
5168 <bch|pmn> * = {200,300},
5169 <blg>    * = {150,200},
5170 <cmr|pad> * = {300,300},
5171 <m-t|cmr|ppl|ptm> + = {250,250},
5172 <bch>    + = {150,250},
5173 <pad>    + = {300,300},
5174 <blg|pmn> + = {150,200},
5175 <ugm>    + = {250,300},
5176 <blg|ugm> {=}= {200,200},

```

```

5177 <m-t|pad|pmn|ptm> ( = {100, }, ) = { ,200},
5178 <bch|ugm> ( = {200, }, ) = { ,200},
5179 <cmr|blg> ( = {300, }, ) = { ,300},
5180 <ppl> ( = {100, }, ) = { ,300},
5181 <bch|pmn> [ = {100, }, ] = { ,100},
5182 <blg> [ = {300,100}, ] = { ,300},

5183 <m-t|pad|pmn|ptm> / = {100,200},
5184 <bch> / = { ,200},
5185 <blg> / = {300,300},
5186 <cmr|ppl> / = {200,300},
5187 <ugm> / = {100,300},
5188 <m-t|ptm> - = {500,500},
5189 <bch|cmr|ppl> - = {400,500},
5190 <blg> - = {300,400},
5191 <pad> - = {300,500},
5192 <pmn> - = {200,400},
5193 <ugm> - = {500,600},
5194 <blg> < = {200,100}, > = {100,200},
5195 <blg> - = {150,250},
5196 <blg> | = {250,250},
5197 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
5198 <bch> \textendash = {200,300}, \textendash = {150,250},
5199 <cmr> \textendash = {400,300}, \textendash = {300,200},
5200 <pad|ppl|ptm> \textendash = {300,300}, \textendash = {200,200},
5201 <ugm> \textendash = {250,300}, \textendash = {250,250},

```

Why settings for left *and* right quotes? Because in some languages they might be used like that (see the csquotes package for examples).

```

5202 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
5203 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
5204 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
5205 <pad|ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
5206 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
5207 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
5208 <m-t|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
5209 <blg> \textquotedblright = {300,400}
5210 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
5211 <pad|ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5212 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
5213 }
5214

```

Greek uppercase letters are in OT1 encoding only.

```

5215 <*m-t|cmr|pmn>
5216 \SetProtrusion
5217 <m-t> [ name = OT1-default,
5218 <cmr> [ name = cmr-OT1,
5219 <pmn> [ name = pmnj-OT1,
5220 <m-t> load = default ]
5221 <cmr> load = cmr-default ]
5222 <pmn> load = pmnj-default ]
5223 <m-t> { encoding = OT1 }
5224 <cmr> { encoding = {OT1,OT4},
5225 <pmn> { encoding = OT1,
5226 <cmr> family = cmr }
5227 <pmn> family = pmnj }
5228 {
5229 <m-t|cmr> \AE = {50, },
5230 <pmn> \OE = {50, }
5231 <*cmr>
5232 "00 = { ,150}, % \Gamma
5233 "01 = {100,100}, % \Delta
5234 "02 = { 50, 50}, % \Theta
5235 "03 = {100,100}, % \Lambda

```

```

5236 "06 = { 50, 50}, % \Sigma
5237 "07 = {100,100}, % \Upsilon
5238 "08 = { 50, 50}, % \Phi
5239 "09 = { 50, 50} % \Psi

```

Remaining slots can be found in the source file.

```

5240 </cmr>
5241 }
5242
5243 </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).

```

5244 \SetProtrusion
5245 <m-t> [ name = T1-default,
5246 <bch> [ name = bch-T1,
5247 <blg> [ name = blg-T1,
5248 <cmr> [ name = cmr-T1,
5249 <pad> [ name = pad-T1,
5250 <pmn> [ name = pmnj-T1,
5251 <ppl> [ name = ppl-T1,
5252 <ptm> [ name = ptm-T1,
5253 <ugm> [ name = ugm-T1,
5254 <m-t> load = default ]
5255 <bch> load = bch-default ]
5256 <blg> load = blg-default ]
5257 <cmr> load = cmr-default ]
5258 <pad> load = pad-default ]
5259 <pmn> load = pmnj-default ]
5260 <ppl> load = ppl-default ]
5261 <ptm> load = ptm-default ]
5262 <ugm> load = ugm-default ]
5263 <m-t> { encoding = {T1,LY1,EU1,EU2} }
5264 <bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
5265 <blg|ptm|ugm> { encoding = {T1},
5266 <bch> family = bch }
5267 <blg> family = blg }
5268 <cmr> family = cmr }
5269 <pad> family = {pad,padx,padj} }
5270 <pmn> family = pmnj }
5271 <ppl> family = {ppl,pplx,pplj} }
5272 <ptm> family = {ptm,ptmx,ptmj} }
5273 <ugm> family = ugm }
5274 {
5275 <m-t|cmr> \AE = {50, },
5276 <bch|pmn> \OE = {50, },
5277 <pmn> \TH = { ,50},
5278 <blg> \v L = { ,250},
5279 <blg> \v d = { ,250},
5280 <blg> \v l = { ,250},
5281 <blg> \v t = { ,250},
5282 <blg> 127 = {300,400},
5283 <blg> 156 = {100, }, % IJ
5284 <blg> 188 = { 80, 80}, % ij
5285 <m-t|bch|pad|pmn|ppl|ptm> _ = {100,100},
5286 <cmr> _ = {200,200},
5287 <ugm> _ = {100,200},
5288 <m-t|pad|pmn|ptm> \textbackslash = {100,200},
5289 <bch> \textbackslash = {150,200},
5290 <blg> \textbackslash = {250,300},
5291 <cmr|ppl> \textbackslash = {200,300},
5292 <ugm> \textbackslash = {100,300},
5293 <ugm> \textbar = {200,200},
5294 <blg> \textendash = {300,300}, \textemdash = {150,150},

```

```

5295 <big> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
5296 <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.

```

5297 <m-t|cmr|pad|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5298 <big> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
5299 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5300 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5301 <big> \guilsinglleft = {300,500}, \guilsinglright = {300,500},
5302 <cmr|pad|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5303 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
5304 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5305 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5306 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
5307 <big|pad|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5308 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
5309 <m-t|bch|cmr|pad|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
5310 <big> \textexclamdown = {200, }, \textquestiondown = {100, },
5311 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
5312 <m-t|cmr|pad|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
5313 <bch|big|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
5314 <m-t|bch|cmr|pad|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
5315 <pmn> \textless = {100, }, \textgreater = { ,100},
5316 <pmn> \textvisiblespace = {100,100} % not in LY1
5317 }
5318

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

5319 <*cmr>
5320 \SetProtrusion
5321 [ name = lmr-T1,
5322   load = cmr-T1 ]
5323 { encoding = {T1,LY1},
5324   family = lmr }
5325 {
5326   \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5327 }
5328
5329 </cmr>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).²¹

```

5330 <*m-t|cmr|pmn>
5331 \SetProtrusion
5332 <m-t> [ name = T2A-default,
5333 <cmr> [ name = cmr-T2A,
5334 <pmn> [ name = pmnj-T2A,
5335 <m-t> load = default ]
5336 <cmr> load = cmr-default ]
5337 <pmn> load = pmnj-default ]
5338 { encoding = T2A,
5339 <m-t> }
5340 <cmr> family = cmr }
5341 <pmn> family = pmnj }
5342 {
5343   \CYRA = {50,50},
5344   \CYRG = { ,50},
5345   \CYRK = { ,50},
5346   \CYRT = {50,50},
5347   \CYRH = {50,50},
5348   \CYRU = {50,50},

```

21 Contributed by Karl Karlsson.

```

5349 <pmn> \CYRS = {50, },
5350 <pmn> \CYRO = {50,50},
5351 \cyrk = { ,50},
5352 \cyrh = { ,50},
5353 \cyrh = {50,50},
5354 <m-t|pmn> \cyru = {50,50},
5355 <cmr> \cyru = {50,70},
5356 <m-t> - = {100,100},
5357 <cmr> - = {200,200},
5358 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,400},
5359 <cmr> \textbackslash = {200,300}, \quotedblbase = {400,400},
5360 <pmn> \textbackslash = {100,200}, \quotedblbase = {300,300},
5361 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},
5362 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5363 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5364 <pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
5365 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5366 <pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
5367 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
5368 <pmn> \textless = {100, }, \textgreater = { ,100}
5369 }
5370
5371 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).²² It also includes some glyphs otherwise in TS1.

```

5372 <*m-t|ptm>
5373 \SetProtrusion
5374 <m-t> [ name = QX-default,
5375 <ptm> [ name = ptm-QX,
5376 <m-t> load = default ]
5377 <ptm> load = ptm-default ]
5378 <m-t> { encoding = QX }
5379 <ptm> { encoding = QX,
5380 <ptm> family = {ptm,ptmx,ptmj} }
5381 {
5382 \AE = {50, },
5383 <ptm> * = {200,200},
5384 {=} = {100,100},
5385 \textunderscore = {100,100},
5386 \textbackslash = {100,200},
5387 \quotedblbase = {400,400},
5388 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5389 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5390 \textexclamdown = {100, }, \textquestiondown = {100, },
5391 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
5392 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
5393 \textless = {200,100}, \textgreater = {100,200},
5394 \textminus = {200,200}, \textdegree = {300,300},
5395 <m-t> \copyright = {100,100}, \textregistered = {100,100}
5396 <ptm> \copyright = {100,150}, \textregistered = {100,150},
5397 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
5398 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
5399 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
5400 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
5401 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
5402 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
5403 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
5404 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
5405 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
5406 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
5407 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
5408 <ptm> \textperthousand = { ,50}

```

22 Contributed by Maciej Eder.

```
5409 }
5410
5411 </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.

```
5412 <*cmr|bch>
5413 \SetProtrusion
5414 <cmr> [ name = cmr-T5,
5415 <cmr> load = cmr-default ]
5416 <bch> [ name = bch-T5,
5417 <bch> load = bch-default ]
5418 { encoding = T5,
5419 <cmr> family = cmr }
5420 <bch> family = bch }
5421 {
5422 <bch> _ = {100,100},
5423 <bch> \textbackslash = {150,200},
5424 <cmr> \textbackslash = {200,300},
5425 <cmr> \textquotedblleft = {200,600},
5426 <cmr> \textquotedbl = {300,300},
5427 <bch> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5428 <cmr> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5429 <bch> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5430 <cmr> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5431 <bch> \guillemotleft = {200,200}, \guillemotright = {150,300},
5432 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5433 <bch> \textbraceleft = {200, }, \textbraceright = { ,300},
5434 <cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5435 \textless = {200,100}, \textgreater = {100,200}
5436 }
5437
5438 </cmr|bch>
```

Minion with lining numbers.

```
5439 <*pmn>
5440 \SetProtrusion
5441 [ name = pmnx-OT1,
5442 load = pmnj-default ]
5443 { encoding = OT1,
5444 family = pmnx }
5445 {
5446 1 = {230,180}
5447 }
5448
5449 \SetProtrusion
5450 [ name = pmnx-T1,
5451 load = pmnj-T1 ]
5452 { encoding = {T1,LY1},
5453 family = pmnx }
5454 {
5455 1 = {230,180}
5456 }
5457
5458 \SetProtrusion
5459 [ name = pmnx-T2A,
5460 load = pmnj-T2A ]
5461 { encoding = {T2A},
5462 family = pmnx }
5463 {
5464 1 = {230,180}
5465 }
5466
5467 </pmn>
```


Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

5468 <*ptm>
5469 \SetProtrusion
5470 [ name = ptm-LY1,
5471   load = ptm-T1 ]
5472 { encoding = LY1,
5473   family = {ptm,ptmx,ptmj} }
5474 {
5475   - = {100,100},
5476   \texttrademark = {100,100},
5477   \textregistered = {100,100},
5478   \textcopyright = {100,100},
5479   \textdegree = {300,300},
5480   \textminus = {200,200},
5481   \textellipsis = {150,200},
5482 % \texteuro = { , }, % ?
5483   \textcent = {100,100},
5484   \textquotesingle = {500,500},
5485   \textflorin = { 50, 70},
5486   \textdagger = {150,150},
5487   \textdaggerdbl = {100,100},
5488   \textperthousand = { , 50},
5489   \textbullet = {150,150},
5490   \textonesuperior = {100,100},
5491   \texttwosuperior = { 50, 50},
5492   \textthreesuperior = { 50, 50},
5493   \textperiodcentered = {300,300},
5494   \textplusminus = { 50, 80},
5495   \textmultiply = {100,100},
5496   \textdivide = { 50,150}

```

Remaining slots in the source file.

```

5497   }
5498
5499 </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.²³

```

5500 \SetProtrusion
5501 <m-t> [ name = OT1-it ]
5502 <bch> [ name = bch-it ]
5503 <blg> [ name = blg-it,
5504 <blg>   load = blg-default ]
5505 <cmr> [ name = cmr-it ]
5506 <pad> [ name = pad-it ]
5507 <pmn> [ name = pmn-it ]
5508 <ppl> [ name = ppl-it ]
5509 <ptm> [ name = ptm-it ]
5510 <ugm> [ name = ugm-it ]
5511 <m-t|bch|blg|pad|ugm> { encoding = OT1,
5512 <ppl|ptm> { encoding = {OT1,OT4},
5513 <bch>   family = bch,
5514 <blg>   family = blg,

```

23 Settings contributed by *Hendrik Vogt*.

```

5515 <pad> family = {pad,padx,padj},
5516 <ppl> family = {ppl,pplx,pplj},
5517 <ptm> family = {ptm,ptmx,ptmj},
5518 <ugm> family = ugm,
5519 <m-t|bch|pad|ppl|ptm> shape = {it,sl} }
5520 <big|ugm> shape = it }
5521 <cmr|pmn> { }
5522 {
5523 <cmr> A = {100,100},
5524 <ptm> A = {100,50},
5525 <pad|pmn> A = {50, },
5526 <ugm> A = { ,150},
5527 <ppl> A = {50,50},
5528 <ptm> \AE = {100, },
5529 <pad|ppl> \AE = {50, },
5530 <cmr> B = {83,-40},
5531 <pad|ppl|ptm> B = {50, },
5532 <pmn> B = {20,-50},
5533 <bch|ppl|ptm|ugm> C = {50, },
5534 <cmr> C = {165,-75},
5535 <pad> C = {100, },
5536 <pmn> C = {50,-50},
5537 <cmr> D = {75, -28},
5538 <pad|ppl|ptm> D = {50,50},
5539 <pmn> D = {20, },
5540 <cmr> E = {80,-55},
5541 <pad|ppl|ptm> E = {50, },
5542 <pmn> E = {20,-50},
5543 <cmr> F = {85,-80},
5544 <pad|ptm> F = {100, },
5545 <pmn> F = {10, },
5546 <ppl> F = {50, },
5547 <bch|ppl|ptm|ugm> G = {50, },
5548 <cmr> G = {153,-15},
5549 <pad> G = {100, },
5550 <pmn> G = {50,-50},
5551 <cmr> H = {73,-60},
5552 <pad|ppl|ptm> H = {50, },
5553 <cmr> I = {140,-120},
5554 <pad|ptm> I = {50, },
5555 <pmn> I = {20,-50},
5556 <cmr> J = {135,-80},
5557 <pad> J = {50, },
5558 <pmn> J = {20, },
5559 <ptm> J = {100, },
5560 <cmr> K = {70,-30},
5561 <pad|ppl|ptm> K = {50, },
5562 <pmn> K = {20, },
5563 <cmr> L = {87, 40},
5564 <pad|ppl|ptm> L = {50, },
5565 <pmn> L = {20,50},
5566 <ugm> L = { ,100},
5567 <cmr> M = {67,-45},
5568 <pmn> M = { , -30},
5569 <ptm> M = {50, },
5570 <cmr> N = {75,-55},
5571 <pmn> N = { , -30},
5572 <ptm> N = {50, },
5573 <bch|pmn|ppl|ptm> O = {50, },
5574 <cmr> O = {150,-30},
5575 <pad> O = {100, },
5576 <ugm> O = {70,50},
5577 <ppl|ptm> \OE = {50, },
5578 <pad> \OE = {100, },
5579 <cmr> P = {82,-50},

```

```

5580 <pad|ppl|ptm>    P = {50, },
5581 <pmn>            P = {20,-50},
5582 <bch|pmn|ppl|ptm> Q = {50, },
5583 <cmr>            Q = {150,-30},
5584 <pad>            Q = {100, },
5585 <ugm>            Q = {70,50},
5586 <cmr>            R = {75, 15},
5587 <pad|ppl|ptm>    R = {50, },
5588 <pmn>            R = {20, },
5589 <bch|pad|ppl|ptm> S = {50, },
5590 <cmr>            S = {90,-65},
5591 <pmn>            S = {20,-30},
5592 <bch|pad|ppl|ptm> $ = {50, },
5593 <cmr>            $ = {100,-20},
5594 <pmn>            $ = {20,-30},
5595 <bch|pmn|ugm>     T = {70, },
5596 <cmr>            T = {220,-85},
5597 <pad|ppl|ptm>    T = {100, },
5598 <cmr>            U = {230,-55},
5599 <pad|ppl|ptm>    U = {50, },
5600 <pmn>            U = {50,-50},
5601 <cmr>            V = {260,-60},
5602 <pad|pmn|ugm>    V = {100, },
5603 <ppl|ptm>         V = {100,50},
5604 <cmr>            W = {185,-55},
5605 <pad|pmn|ugm>    W = {100, },
5606 <ppl>            W = {50, },
5607 <ptm>            W = {100,50},
5608 <cmr>            X = {70,-30},
5609 <ppl|ptm>        X = {50, },
5610 <cmr>            Y = {250,-60},
5611 <pmn>            Y = {50, },
5612 <ppl>            Y = {100,50},
5613 <ptm>            Y = {100, },
5614 <cmr>            Z = {90,-60},
5615 <pmn>            Z = { , -50},
5616 <cmr>            a = {150,-10},
5617 <cmr>            b = {170, },
5618 <cmr>            c = {173,-10},
5619 <cmr>            d = {150,-55},
5620 <pmn>            d = { , -50},
5621 <cmr>            e = {180, },
5622 <cmr>            f = { , -250},
5623 <pad|pmn>        f = { , -100},
5624 <cmr>            g = {150,-10},
5625 <cmr>            h = {100, },
5626 <cmr>            i = {210, },
5627 <pmn>            i = { , -30},
5628 <cmr>            j = { , -40},
5629 <pmn>            j = { , -30},
5630 <cmr>            k = {110,-50},
5631 <cmr>            l = {240,-110},
5632 <pmn>            l = { , -100},
5633 <cmr>            m = {80, },
5634 <cmr>            n = {115, },
5635 <bch>            o = {50,50},
5636 <cmr>            o = {155, },
5637 <bch>            p = { , 50},
5638 <pmn>            p = {-50, },
5639 <bch>            q = {50, },
5640 <cmr>            q = {170,-40},
5641 <cmr>            r = {155,-40},
5642 <pmn>            r = { , 50},
5643 <cmr>            s = {130, },
5644 <bch>            t = { , 50},

```

```

5645 <cmr>      t = {230,-10},
5646 <cmr>      u = {120, },
5647 <cmr>      v = {140,-25},
5648 <pmn|ugm>  v = {50, },
5649 <bch>      w = { ,50},
5650 <cmr>      w = {98,-20},
5651 <pmn|ugm>  w = {50, },
5652 <cmr>      x = {65,-40},
5653 <bch>      y = { ,50},
5654 <cmr>      y = {130,-20},
5655 <cmr>      z = {110,-80},
5656 <cmr>      0 = {170,-85},
5657 <bch|ptm>  1 = {150,100},
5658 <cmr>      1 = {230,110},
5659 <pad>      1 = {150, },
5660 <pmn>      1 = {50, },
5661 <ppl>      1 = {100, },
5662 <ugm>      1 = {150,150},
5663 <cmr>      2 = {130,-70},
5664 <pad|ppl|ptm>  2 = {50, },
5665 <pmn>      2 = {-50, },
5666 <bch>      3 = {50, },
5667 <cmr>      3 = {140,-70},
5668 <pmn>      3 = {-100, },
5669 <ptm>      3 = {100,50},
5670 <bch>      4 = {100, },
5671 <cmr>      4 = {130,80},
5672 <pad>      4 = {150, },
5673 <ppl|ptm>  4 = {50, },
5674 <cmr>      5 = {160, },
5675 <ptm>      5 = {50, },
5676 <bch>      6 = {50, },
5677 <cmr>      6 = {175,-30},
5678 <bch|pad|ptm>  7 = {100, },
5679 <cmr>      7 = {250,-150},
5680 <pmn>      7 = {20, },
5681 <ppl>      7 = {50, },
5682 <cmr>      8 = {130,-40},
5683 <cmr>      9 = {155,-80},
5684 <m-t|cmr|pad|pmn|ppl>  . = { ,500},
5685 <big>      . = {400,600},
5686 <bch|ptm|ugm>  . = { ,700},
5687 <big>      {,}= {300,500},
5688 <m-t|pad|pmn|ppl>  {,}= { ,500},
5689 <cmr>      {,}= { ,450},
5690 <bch|ugm>      {,}= { ,600},
5691 <ptm>      {,}= { ,700},
5692 <m-t|cmr|pad|ppl>  : = { ,300},
5693 <bch|ugm>      : = { ,400},
5694 <pmn>      : = { ,200},
5695 <ptm>      : = { ,500},
5696 <m-t|cmr|pad|ppl>  ; = { ,300},
5697 <bch|ugm>      ; = { ,400},
5698 <pmn>      ; = { ,200},
5699 <ptm>      ; = { ,500},
5700 <ptm>      ! = { ,100},
5701 <bch>      ? = { ,200},
5702 <ptm>      ? = { ,100},
5703 <ppl>      ? = { ,300},
5704 <pmn>      " = {400,200},
5705 <m-t|pad|pmn|ppl|ptm>  & = {50,50},
5706 <bch>      & = { ,80},
5707 <cmr>      & = {130,30},
5708 <ugm>      & = {50,100},
5709 <m-t|pad|pmn>      \% = {100, },

```

```

5710 <cmr> \% = {180,50},
5711 <bch> \% = {50,50},
5712 <ppl|ptm> \% = {100,100},
5713 <ugm> \% = {100,50},
5714 <m-t|pmn|ppl> * = {200,200},
5715 <bch> * = {300,200},
5716 <cmr> * = {380,20},
5717 <pad> * = {500,100},
5718 <ptm|ugm> * = {400,200},
5719 <m-t|pmn|ppl> + = {150,200},
5720 <cmr> + = {180,200},
5721 <bch|ugm> + = {250,250},
5722 <pad|ptm> + = {250,200},
5723 <m-t|pad|pmn|ppl> @ = {50,50},
5724 <bch> @ = {80,50},
5725 <cmr> @ = {180,10},
5726 <ptm> @ = {150,150},
5727 <m-t|bch|ugm> ~ = {150,150},
5728 <cmr|pad|pmn|ppl|ptm> ~ = {200,150},
5729 <ugm> {=} = {200,200},
5730 <m-t|bch|pad|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
5731 <cmr> ( = {300, }, ) = { ,70},
5732 <m-t|pad|ppl|ptm|ugm> / = {100,200},
5733 <cmr> / = {100,100},
5734 <bch> / = { ,150},
5735 <pmn> / = {100,150},
5736 <m-t> - = {300,300},
5737 <bch|pad> - = {300,400},
5738 <pmn> - = {200,300},
5739 <cmr> - = {500,300},
5740 <ppl> - = {300,500},
5741 <ptm> - = {500,500},
5742 <ugm> - = {400,700},
5743 <blg> - = {0,300},
5744 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
5745 <bch> \textendash = {200,300}, \textendash = {150,200},
5746 <cmr> \textendash = {500,300}, \textendash = {400,170},
5747 <pad|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
5748 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
5749 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
5750 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
5751 <pad> \textquoteleft = {800,200}, \textquoteright = {800,200},
5752 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
5753 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
5754 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
5755 <blg> \textquotedblright = {300,300}
5756 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
5757 <pad> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
5758 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
5759 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
5760 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
5761 }
5762
5763 <*cmr|pmn>
5764 \SetProtrusion
5765 <cmr> [ name = cmr-it-OT1,
5766 <pmn> [ name = pmnj-it-OT1,
5767 <cmr> load = cmr-it ]
5768 <pmn> load = pmnj-it ]
5769 <cmr> { encoding = {OT1,OT4},
5770 <pmn> { encoding = OT1,
5771 <cmr> family = cmr,
5772 <pmn> family = pmnj,
5773 <cmr> shape = it }
5774 <pmn> shape = {it,s1} }

```

```

5775 {
5776 <cmr> \AE = {100, },
5777 <pmn> \AE = { , -50},
5778 <cmr> \OE = {100, },
5779 <pmn> \OE = {50, }
5780 <*cmr>
5781 "00 = {200,150}, % \Gamma
5782 "01 = {150,100}, % \Delta
5783 "02 = {150, 50}, % \Theta
5784 "03 = {150, 50}, % \Lambda
5785 "04 = {100,100}, % \Xi
5786 "05 = {100,100}, % \Pi
5787 "06 = {100, 50}, % \Sigma
5788 "07 = {200,150}, % \Upsilon
5789 "08 = {150, 50}, % \Phi
5790 "09 = {150,100}, % \Psi
5791 "0A = { 50, 50} % \Omega
5792 </cmr>
5793 }
5794
5795 </cmr|pmn>
5796 \SetProtrusion
5797 <m-t> [ name = T1-it-default,
5798 <bch> [ name = bch-it-T1,
5799 <blg> [ name = blg-it-T1,
5800 <cmr> [ name = cmr-it-T1,
5801 <pad> [ name = pad-it-T1,
5802 <pmn> [ name = pmnj-it-T1,
5803 <ppl> [ name = ppl-it-T1,
5804 <ptm> [ name = ptm-it-T1,
5805 <ugm> [ name = ugm-it-T1,
5806 <m-t> load = OT1-it ]
5807 <bch> load = bch-it ]
5808 <blg> load = blg-T1 ]
5809 <cmr> load = cmr-it ]
5810 <pmn> load = pmnj-it ]
5811 <pad> load = pad-it ]
5812 <ppl> load = ppl-it ]
5813 <ptm> load = ptm-it ]
5814 <ugm> load = ugm-it ]
5815 <m-t|bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
5816 <blg|ptm|ugm> { encoding = T1,
5817 <bch> family = bch,
5818 <blg> family = blg,
5819 <cmr> family = cmr,
5820 <pmn> family = pmnj,
5821 <pad> family = {pad,padx,padj},
5822 <ppl> family = {ppl,pplx,pplj},
5823 <ptm> family = {ptm,ptmx,ptmj},
5824 <ugm> family = ugm,
5825 <m-t|bch|pad|pmn|ppl|ptm> shape = {it,sl} }
5826 <blg|cmr|ugm> shape = it }
5827 {
5828 <m-t|bch|pmn> _ = { ,100},
5829 <blg> _ = {0,300},
5830 <cmr|ugm> _ = {100,200},
5831 <pad|ppl|ptm> _ = {100,100},
5832 <blg> . = {400,600},
5833 <blg> {,}= {300,500},
5834 <cmr> \AE = {100, },
5835 <pmn> \AE = { , -50},
5836 <bch|pmn> \OE = { 50, },
5837 <cmr> \OE = {100, },
5838 <pmn> 031 = { , -100}, % ff1
5839 <cmr|ptm> 156 = {100, }, % IJ

```

```

5840 <pad> 156 = {50, }, % IJ
5841 <pmn> 156 = {20, }, % IJ
5842 <pmn> 188 = { , -30}, % ij
5843 <pmn> \v t = { , 100},
5844 <m-t|pad|ppl|ptm> \textbackslash = {100,200},
5845 <cmr|ugm> \textbackslash = {300,300},
5846 <bch> \textbackslash = {150,150},
5847 <pmn> \textbackslash = {100,150},
5848 <ugm> \textbar = {200,200},
5849 <cmr> \textquotedblleft = {500,300},
5850 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
5851 <blg> \textquotedbl = {300,300}, \textquotedblleft = {300,300},
5852 <blg> \textquotedblright = {300,300}, \quotedblbase = {200,600},
5853 <m-t|ptm> \quotesinglbase = {300,700}, \quotedblbase = {400,500},
5854 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
5855 <bch|pmn> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
5856 <pad|ppl> \quotesinglbase = {500,500}, \quotedblbase = {400,400},
5857 <ugm> \quotesinglbase = {300,700}, \quotedblbase = {300,500},
5858 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5859 <bch|pmn> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
5860 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
5861 <pad> \guilsinglleft = {500,400}, \guilsinglright = {300,500},
5862 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
5863 <m-t|ppl> \guillemotleft = {300,300}, \guillemotright = {300,300},
5864 <bch|pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
5865 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
5866 <pad> \guillemotleft = {300,300}, \guillemotright = {200,400},
5867 <ptm> \guillemotleft = {300,400}, \guillemotright = {200,400},
5868 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
5869 <m-t|pad|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
5870 <cmr|ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
5871 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
5872 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
5873 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { , 200},
5874 <cmr|pad|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
5875 <bch|pmn> \textless = {100, }, \textgreater = { , 100},
5876 <cmr|pad|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
5877 <pmn> \textvisiblespace = {100,100}
5878 }
5879
5880 <*m-t|cmr|pmn>
5881 \SetProtrusion
5882 <m-t> [ name = T2A-it-default,
5883 <cmr> [ name = cmr-it-T2A,
5884 <pmn> [ name = pmnj-it-T2A,
5885 <m-t> load = OT1-it ]
5886 <cmr> load = cmr-it ]
5887 <pmn> load = pmnj-it ]
5888 { encoding = T2A,
5889 <cmr> family = cmr,
5890 <pmn> family = pmnj,
5891 <m-t|pmn> shape = {it,sl} }
5892 <cmr> shape = it }
5893 {
5894 <cmr> \CYRA = {100,50},
5895 <pmn> \CYRA = {50, },
5896 <cmr> \CYRB = {50, },
5897 <cmr> \CYRV = {50, },
5898 <pmn> \CYRV = {20,-50},
5899 <cmr> \CYRG = {100, },
5900 <pmn> \CYRG = {10, },
5901 <cmr> \CYRD = {50, },
5902 <cmr> \CYRE = {50, },
5903 <pmn> \CYRE = {20,-50},
5904 <cmr> \CYRZH = {50, },

```

```

5905 <cmr> \CYRZ = {50, },
5906 <pmn> \CYRZ = {20,-50},
5907 <cmr> \CYRI = {50, },
5908 <pmn> \CYRI = { , -30},
5909 <cmr> \CYRISHRT = {50, },
5910 <cmr> \CYRK = {50, },
5911 <pmn> \CYRK = {20, },
5912 <cmr> \CYRL = {50, },
5913 <cmr> \CYRM = {50, },
5914 <pmn> \CYRM = { , -30},
5915 <cmr> \CYRN = {50, },
5916 <cmr> \CYRO = {100, },
5917 <pmn> \CYRO = {50, },
5918 <cmr> \CYRP = {50, },
5919 <cmr> \CYRR = {50, },
5920 <pmn> \CYRR = {20,-50},
5921 <cmr> \CYRS = {100, },
5922 <pmn> \CYRS = {50, },
5923 <cmr> \CYRT = {100, },
5924 <pmn> \CYRT = {70, },
5925 <cmr> \CYRU = {100, },
5926 <pmn> \CYRU = {50, },
5927 <cmr> \CYRF = {100, },
5928 <cmr> \CYRH = {50, },
5929 <cmr> \CYRC = {50, },
5930 <cmr> \CYRCH = {100, },
5931 <cmr> \CYRSH = {50, },
5932 <cmr> \CYRSHCH = {50, },
5933 <cmr> \CYRHRDSN = {100, },
5934 <cmr> \CYRERY = {50, },
5935 <cmr> \CYRSFTSN = {50, },
5936 <cmr> \CYREREV = {50, },
5937 <cmr> \CYRYU = {50, },
5938 <cmr> \CYRYA = {50, },
5939 <pmn> \CYRYA = { , 20},
5940 <pmn> \cyrr = {-50, },
5941 <m-t|pmn> _ = { , 100},
5942 <cmr> _ = {100,200},
5943 <pmn> 031 = { , -100}, % ff1
5944 <pmn> \v t = { , 100},
5945 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
5946 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
5947 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
5948 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
5949 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
5950 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
5951 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
5952 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
5953 <pmn> \textbraceleft = {200, }, \textbraceright = { , 200},
5954 <cmr> \textquotedblleft = {500,300},
5955 <cmr> \textless = {300,100}, \textgreater = {200,100}
5956 <pmn> \textless = {100, }, \textgreater = { , 100}
5957 }
5958
5959 </m-t|cmr|pmn>
5960 <*m-t|ptm>
5961 \SetProtrusion
5962 <m-t> [ name = QX-it-default,
5963 <ptm> [ name = ptm-it-QX,
5964 <m-t> load = OT1-it ]
5965 <ptm> load = ptm-it ]
5966 { encoding = {QX},
5967 <ptm> family = {ptm,ptmx,ptmj},
5968 shape = {it,s1} }
5969 {

```



```

5970 <ptm>    009 = { , 50}, % fk
5971      {=} = {100,100},
5972 <m-t>    \textunderscore = {100,100},
5973 <ptm>    \textunderscore = {100,150},
5974      \textbackslash = {100,200},
5975      \quotedblbase = {300,400},
5976 <m-t>    \guillemotleft = {300,300}, \guillemotright = {300,300},
5977 <ptm>    \guillemotleft = {200,400}, \guillemotright = {200,400},
5978      \textexclamdown = {200, }, \textquestiondown = {200, },
5979      \textbraceleft = {200,100}, \textbraceright = {200,200},
5980      \textless = {100,100}, \textgreater = {100,100},
5981      \textminus = {200,200}, \textdegree = {300,150},
5982 <m-t>    \copyright = {100,100}, \textregistered = {100,100}
5983 <ptm>    \textregistered = {100,150}, \copyright = {100,150},
5984 <ptm>    \textDelta = { 70, }, \textdelta = { , 50},
5985 <ptm>    \textpi = { 50, 80}, \textmu = { , 80},
5986 <ptm>    \texteuro = {200, }, \textellipsis = {100,200},
5987 <ptm>    \textquoteleft = {500,400}, \textquoteright = {500,400},
5988 <ptm>    \textquotedblleft = {500,300}, \textquotedblright = {400,400},
5989 <ptm>    \textapprox = { 50, 50}, \textinfty = {100,100},
5990 <ptm>    \textdagger = {150,150}, \textdaggerdbl = {100,100},
5991 <ptm>    \textdiv = {150,150}, \textasciitilde = { 80, 80},
5992 <ptm>    \texttimes = {100,150}, \textpm = { 50, 80},
5993 <ptm>    \textbullet = {300,100}, \textperiodcentered = {300,300},
5994 <ptm>    \textquotesingle = {500,500}, \textquotedbl = {300,300},
5995 <ptm>    \textperthousand = { ,50}
5996   }
5997
5998 </m-t|ptm>
5999 <*cmr|bch>
6000 \SetProtrusion
6001 <cmr> [ name = cmr-it-T5,
6002 <cmr>   load = cmr-it ]
6003 <bch> [ name = bch-it-T5,
6004 <bch>   load = bch-it ]
6005   { encoding = T5,
6006 <bch>   family = bch,
6007 <cmr>   family = cmr,
6008     shape = it }
6009   {
6010 <bch>     _ = { ,100},
6011 <cmr>     _ = {100,200},
6012 <bch>     \textbackslash = {150,150},
6013 <cmr>     \textbackslash = {300,300},
6014 <bch>     \quotesinglbase = {200,500}, \quotedblbase = {150,500},
6015 <cmr>     \quotesinglbase = {300,700}, \quotedblbase = {200,600},
6016 <bch>     \guilsinglleft = {300,400}, \guilsinglright = {200,500},
6017 <cmr>     \guilsinglleft = {500,300}, \guilsinglright = {400,400},
6018 <bch>     \guillemotleft = {200,300}, \guillemotright = {150,400},
6019 <cmr>     \guillemotleft = {400,100}, \guillemotright = {200,300},
6020 <bch>     \textbraceleft = {200, }, \textbraceright = { ,200},
6021 <cmr>     \textbraceleft = {400,100}, \textbraceright = {200,200},
6022 <bch>     \textless = {100, }, \textgreater = { ,100}
6023 <cmr>     \textless = {300,100}, \textgreater = {200,100}
6024   }
6025
6026 </cmr|bch>

```

Slanted is very similar to italic.

```

6027 <*cmr>
6028 \SetProtrusion
6029 [ name = cmr-sl,
6030   load = cmr-it-OT1 ]
6031 { encoding = {OT1,OT4},
6032   family = cmr,

```

```

6033     shape    = sl }
6034     {
6035         L = { ,50},
6036         f = { ,-50},
6037         - = {300, },
6038         \textendash = {400, }, \textemdash = {300, }
6039     }
6040
6041 \SetProtrusion
6042 [ name    = cmr-sl-T1,
6043   load    = cmr-it-T1 ]
6044 { encoding = {T1,LY1},
6045   family  = cmr,
6046   shape   = sl }
6047 {
6048     L = { ,50},
6049     f = { ,-50},
6050     - = {300, },
6051     \textendash = {400, }, \textemdash = {300, }
6052 }
6053
6054 \SetProtrusion
6055 [ name    = cmr-sl-T2A,
6056   load    = cmr-it-T2A ]
6057 { encoding = T2A,
6058   family  = cmr,
6059   shape   = sl }
6060 {
6061     L = { ,50},
6062     f = { ,-50},
6063     - = {300, },
6064     \textendash = {400, }, \textemdash = {300, }
6065 }
6066
6067 \SetProtrusion
6068 [ name    = cmr-sl-T5,
6069   load    = cmr-it-T5 ]
6070 { encoding = T5,
6071   family  = cmr,
6072   shape   = sl }
6073 {
6074     L = { ,50},
6075     f = { ,-50},
6076     - = {300, },
6077     \textendash = {400, }, \textemdash = {300, }
6078 }
6079
6080 \SetProtrusion
6081 [ name    = lmr-it-T1,
6082   load    = cmr-it-T1 ]
6083 { encoding = {T1,LY1},
6084   family  = lmr,
6085   shape   = {it,sl} }
6086 {
6087     \textquotedblleft = { ,200}, \textquotedblright = { ,200},
6088     \quotesinglbase   = { ,400}, \quotedblbase     = { ,500}
6089 }
6090

```

Oldstyle numerals are slightly different.

```

6091 \SetProtrusion
6092 [ name = cmr(oldstyle)-it,
6093   load = cmr-it-T1 ]
6094 { encoding = T1,
6095   family  = {hfor,cmor},

```

```

6096     shape    = {it,s1} }
6097     {
6098       1 = {250, 50},
6099       2 = {150,-100},
6100       3 = {100,-50},
6101       4 = {150,150},
6102       6 = {200,  },
6103       7 = {200, 50},
6104       8 = {150,-50},
6105       9 = {100, 50}
6106     }
6107
6108 </cmr>
6109 < *pmn >
6110 \SetProtrusion
6111   [ name    = pmnx-it,
6112     load    = pmnj-it ]
6113   { encoding = OT1,
6114     family   = pmnx,
6115     shape    = {it,s1} }
6116   {
6117     1 = {100,150}
6118   }
6119
6120 \SetProtrusion
6121   [ name    = pmnx-it-T1,
6122     load    = pmnj-it-T1 ]
6123   { encoding = {T1,LY1},
6124     family   = pmnx,
6125     shape    = {it,s1} }
6126   {
6127     1 = {100,150}
6128   }
6129
6130 \SetProtrusion
6131   [ name    = pmnx-it-T2A,
6132     load    = pmnj-it-T2A ]
6133   { encoding = {T2A},
6134     family   = pmnx,
6135     shape    = {it,s1} }
6136   {
6137     1 = {100,150}
6138   }
6139
6140 </pmn >
6141 < *ptm >
6142 \SetProtrusion
6143   [ name    = ptm-it-LY1,
6144     load    = ptm-it-T1 ]
6145   { encoding = {LY1},
6146     family   = {ptm,ptmx,ptmj},
6147     shape    = {it,s1} }
6148   {
6149     -                               = {100,100},
6150     \texttrademark                 = {100,100},
6151     \textregistered                 = {100,100},
6152     \textcopyright                  = {100,100},
6153     \textdegree                     = {300,100},
6154     \textminus                      = {200,200},
6155     \textellipsis                   = {100,200},
6156     % \texteuro                     = {  ,  }, % ?
6157     \textcent                       = {100,100},
6158     \textquotesingle                = {500,  },
6159     \textflorin                     = {100, 70},
6160     \textdagger                     = {150,150},

```

```

6161 \textdaggerdbl = {100,100},
6162 \textbullet = {150,150},
6163 \textonesuperior = {150,100},
6164 \texttwosuperior = {150, 50},
6165 \textthreesuperior = {150, 50},
6166 \textparagraph = {100, },
6167 \textperiodcentered = {500,300},
6168 \textonequarter = { 50, },
6169 \textonehalf = { 50, },
6170 \textplusminus = {100,100},
6171 \textmultiply = {150,150},
6172 \textdivide = {150,150}
6173 }
6174
6175 </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).

```

6176 <*(blg|ugm)>
6177 \SetProtrusion
6178 <m-t> [ name = OT1-sc,
6179 <bch> [ name = bch-sc,
6180 <cmr> [ name = cmr-sc-OT1,
6181 <pad> [ name = pad-sc,
6182 <pmn> [ name = pmnj-sc,
6183 <ppl> [ name = ppl-sc,
6184 <ptm> [ name = ptm-sc,
6185 <m-t> load = default ]
6186 <bch> load = bch-default ]
6187 <cmr> load = cmr-OT1 ]
6188 <pad> load = pad-default ]
6189 <pmn> load = pmnj-default ]
6190 <ppl> load = ppl-default ]
6191 <ptm> load = ptm-default ]
6192 <m-t|bch|pad|pmn> { encoding = OT1,
6193 <cmr|ppl|ptm> { encoding = {OT1,OT4},
6194 <bch> family = bch,
6195 <cmr> family = cmr,
6196 <pad> family = {pad,padx,padj},
6197 <pmn> family = pmnj,
6198 <ppl> family = {ppl,pplx,pplj},
6199 <ptm> family = {ptm,ptmx,ptmj},
6200 shape = sc }
6201 {
6202 a = {50,50},
6203 <cmr|pad|ppl|ptm> \ae = {50, },
6204 <bch|pmn> c = {50, },
6205 <bch|pad|pmn> d = { ,50},
6206 <m-t|bch|cmr|pad|pmn|ptm> f = { ,50},
6207 <bch|pad|pmn> g = {50, },
6208 <m-t|cmr|pad|pmn|ppl|ptm> j = {50, },
6209 <bch> j = {100, },
6210 <m-t|bch|cmr|pad|pmn|ppl> l = { ,50},
6211 <ptm> l = { ,80},
6212 <m-t|bch|cmr|pad|pmn|ppl> 013 = { ,50}, % fl
6213 <ptm> 013 = { ,80}, % fl
6214 <bch|pad|pmn> o = {50,50},
6215 <pad|pmn> \oe = {50, },
6216 <ppl> p = { 0, 0},
6217 <bch|pad|pmn> q = {50,70},

```

```

6218 <ppi> q = { 0, },
6219 <m-t|cmr|pad|pmn|ppi|ptm> r = { , 0},
6220 t = {50,50},
6221 <m-t|bch|cmr|pad|pmn|ppi> y = {50,50}
6222 <ptm> y = {80,80}
6223 }
6224
6225 \SetProtrusion
6226 <m-t> [ name = T1-sc,
6227 <bch> [ name = bch-sc-T1,
6228 <cmr> [ name = cmr-sc-T1,
6229 <pad> [ name = pad-sc-T1,
6230 <pmn> [ name = pmnj-sc-T1,
6231 <ppi> [ name = ppl-sc-T1,
6232 <ptm> [ name = ptm-sc-T1,
6233 <m-t> load = T1-default ]
6234 <bch> load = bch-T1 ]
6235 <cmr> load = cmr-T1 ]
6236 <pad> load = pad-T1 ]
6237 <pmn> load = pmnj-T1 ]
6238 <ppi> load = ppl-T1 ]
6239 <ptm> load = ptm-T1 ]
6240 { encoding = {T1,LY1},
6241 <bch> family = bch,
6242 <cmr> family = cmr,
6243 <pad> family = {pad,padx,padj},
6244 <pmn> family = pmnj,
6245 <ppi> family = {ppl,pplx,pplj},
6246 <ptm> family = {ptm,ptmx,ptmj},
6247 shape = sc }
6248 {
6249 a = {50,50},
6250 <cmr|pad|ppi|ptm> \ae = {50, },
6251 <bch|pmn> c = {50, },
6252 <bch|pad|pmn> d = { ,50},
6253 <m-t|bch|cmr|pad|pmn|ptm> f = { ,50},
6254 <bch|pad|pmn> g = {50, },
6255 <m-t|cmr|pad|pmn|ppi|ptm> j = {50, },
6256 <bch> j = {100, },
6257 <m-t|bch|cmr|pad|pmn|ppi> l = { ,50},
6258 <ptm> l = { ,80},
6259 <m-t|bch|cmr|pad|pmn|ppi> 029 = { ,50}, % fl
6260 <ptm> 029 = { ,80}, % fl
6261 <bch|pad|pmn> o = {50,50},
6262 <bch|pad|pmn> \oe = {50, },
6263 <ppi> p = { 0, 0},
6264 <bch|pad|pmn> q = {50,70},
6265 <ppi> q = { 0, },
6266 <m-t|cmr|pad|pmn|ppi|ptm> r = { , 0},
6267 t = {50,50},
6268 <m-t|bch|cmr|pad|pmn|ppi> y = {50,50}
6269 <ptm> y = {80,80}
6270 }
6271
6272 <!(big|ugm)>
6273 <*m-t|cmr>
6274 \SetProtrusion
6275 <m-t> [ name = T2A-sc,
6276 <cmr> [ name = cmr-sc-T2A,
6277 <m-t> load = T2A-default ]
6278 <cmr> load = cmr-T2A ]
6279 { encoding = T2A,
6280 <cmr> family = cmr,
6281 shape = sc }
6282 {

```

```

6283     \cyra = {50,50},
6284     \cyrg = { ,50},
6285     \cyrt = {50,50},
6286     \cyyr = { ,50}
6287   }
6288
6289 </m-t|cmr>
6290 <*m-t>
6291 \SetProtrusion
6292   [ name      = QX-sc,
6293     load      = QX-default ]
6294   { encoding = QX,
6295     shape     = sc }
6296   {
6297     a = {50,50},
6298     f = { ,50},
6299     j = {50, },
6300     l = { ,50},
6301     013 = { ,50}, % fl
6302     r = { , 0},
6303     t = {50,50},
6304     y = {50,50}
6305   }
6306
6307 </m-t>
6308 <*cmr|bch>
6309 \SetProtrusion
6310 <bch> [ name      = bch-sc-T5,
6311 <bch>   load      = bch-T5 ]
6312 <cmr> [ name      = cmr-sc-T5,
6313 <cmr>   load      = cmr-T5 ]
6314   { encoding = T5,
6315 <bch>   family   = bch,
6316 <cmr>   family   = cmr,
6317     shape    = sc }
6318   {
6319     a = {50,50},
6320 <bch>   c = {50, },
6321 <bch>   d = { ,50},
6322     f = { ,50},
6323 <bch>   g = {50, },
6324 <bch>   j = {100, },
6325 <cmr>   j = {50, },
6326     l = { ,50},
6327 <bch>   o = {50,50},
6328 <bch>   q = { 0, },
6329 <cmr>   r = { , 0},
6330     t = {50,50},
6331     y = {50,50}
6332   }
6333
6334 </cmr|bch>
6335 <*pmn>
6336 \SetProtrusion
6337   [ name      = pmnx-sc,
6338     load      = pmnj-sc ]
6339   { encoding = OT1,
6340     family   = pmnx,
6341     shape    = sc }
6342   {
6343     l = {230,180}
6344   }
6345
6346 \SetProtrusion
6347   [ name      = pmnx-sc-T1,

```

```

6348     load      = pmnj-sc-T1 ]
6349     { encoding = {T1,LY1},
6350       family   = pmnx,
6351       shape    = sc }
6352     {
6353       l = {230,180}
6354     }
6355

```

15.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's font installation guide suggests `si`.

```

6356 \SetProtrusion
6357 [ name      = pmnj-scit,
6358   load      = pmnj-it  ]
6359 { encoding = OT1,
6360   family   = pmnj,
6361   shape    = {scit,si} }
6362 {
6363   a = {50, },
6364   \ae = { , -50},
6365   b = {20, -50},
6366   c = {50, -50},
6367   d = {20, 0},
6368   e = {20, -50},
6369   f = {10, 0},
6370   012 = {10, -50}, % fi
6371   013 = {10, -50}, % fl
6372   014 = {10, -50}, % ffi
6373   015 = {10, -50}, % ffl
6374   g = {50, -50},
6375   i = {20, -50},
6376   j = {20, 0},
6377   k = {20, },
6378   l = {20, 50},
6379   m = { , -30},
6380   n = { , -30},
6381   o = {50, },
6382   \oe = {50, -50},
6383   p = {20, -50},
6384   q = {50, },
6385   r = {20, 0},
6386   s = {20, -30},
6387   t = {70, },
6388   u = {50, -50},
6389   v = {100, },
6390   w = {100, },
6391   y = {50, },
6392   z = { , -50}
6393 }
6394
6395 \SetProtrusion
6396 [ name      = pmnj-scit-T1,
6397   load      = pmnj-it-T1 ]
6398 { encoding = {T1,LY1},
6399   family   = pmnj,
6400   shape    = {scit,si} }
6401 {
6402   a = {50, },
6403   \ae = { , -50},
6404   b = {20, -50},
6405   c = {50, -50},
6406   d = {20, 0},

```

```

6407     e = {20,-50},
6408     f = {10, 0},
6409     028 = {10,-50}, % fi
6410     029 = {10,-50}, % fl
6411     030 = {10,-50}, % ffi
6412     031 = {10,-50}, % ff1
6413     g = {50,-50},
6414     i = {20,-50},
6415     188 = {20, 0}, % ij
6416     j = {20, 0},
6417     k = {20, },
6418     l = {20,50},
6419     m = { , -30},
6420     n = { , -30},
6421     o = {50, },
6422     \oe = {50,-50},
6423     p = {20,-50},
6424     q = {50, },
6425     r = {20, 0},
6426     s = {20,-30},
6427     t = {70, },
6428     u = {50,-50},
6429     v = {100, },
6430     w = {100, },
6431     y = {50, },
6432     z = { , -50}
6433 }
6434
6435 \SetProtrusion
6436 [ name = pmnx-scit,
6437   load = pmnj-scit ]
6438 { encoding = OT1,
6439   family = pmnx,
6440   shape = {scit,si} }
6441 {
6442   l = {100,150}
6443 }
6444
6445 \SetProtrusion
6446 [ name = pmnx-scit-T1,
6447   load = pmnj-scit-T1 ]
6448 { encoding = {T1,LY1},
6449   family = pmnx,
6450   shape = {scit,si} }
6451 {
6452   l = {100,150}
6453 }
6454
6455 </pmn>

```

15.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino. Anybody?

```

6456 \SetProtrusion
6457 <m-t> [ name = textcomp ]
6458 <bch> [ name = bch-textcomp ]
6459 <blg> [ name = blg-textcomp ]
6460 <cmr> [ name = cmr-textcomp ]
6461 <pad> [ name = pad-textcomp ]
6462 <pmn> [ name = pmn-textcomp ]
6463 <ppl> [ name = ppl-textcomp ]
6464 <ptm> [ name = ptm-textcomp ]
6465 <ugm> [ name = ugm-textcomp ]

```



```

6466 <m-t> { encoding = TS1      }
6467 <!m-t> { encoding = TS1,
6468 <bch>    family = bch }
6469 <blg>    family = blg }
6470 <cmr>    family = cmr }
6471 <pad>    family = {pad,padx,padj} }
6472 <pmn>    family = {pmnx,pmnj} }
6473 <tpl>    family = {tpl,tplx,tplj} }
6474 <ptm>    family = {ptm,ptmx,ptmj} }
6475 <ugm>    family = ugm }
6476 {
6477 <blg>    \textquotestraightbase = {400,500},
6478 <cmr>    \textquotestraightbase = {300,300},
6479 <pad|pmn> \textquotestraightbase = {400,400},
6480 <blg>    \textquotestraightdblbase = {300,400},
6481 <cmr|pmn> \textquotestraightdblbase = {300,300},
6482 <pad>    \textquotestraightdblbase = {400,400},
6483 <bch|cmr|pad|pmn|ugm> \texttwelveudash = {200,200},
6484 <bch|cmr|pad|pmn> \textthreequartersemdash = {150,150},
6485 <ugm>    \textthreequartersemdash = {200,200},
6486 <blg>    \textquotesingle = {500,600},
6487 <cmr|pmn> \textquotesingle = {300,400},
6488 <pad>    \textquotesingle = {400,500},
6489 <ptm>    \textquotesingle = {500,500},
6490 <ugm>    \textquotesingle = {300,500},
6491 <bch|cmr|pmn> \textasteriskcentered = {200,300},
6492 <blg>    \textasteriskcentered = {150,200},
6493 <pad>    \textasteriskcentered = {300,300},
6494 <ugm>    \textasteriskcentered = {100,200},
6495 <pmn>    \textfractionsolidus = {-200,-200},
6496 <cmr>    \textoneoldstyle = {100,100},
6497 <pmn>    \textoneoldstyle = { , 50},
6498 <cmr>    \textthreeoldstyle = { , 50},
6499 <pad|pmn> \textthreeoldstyle = { 50, },
6500 <cmr>    \textfouroldstyle = { 50, 50},
6501 <pad|pmn> \textfouroldstyle = { 50, },
6502 <cmr|pad|pmn> \textsevenoldstyle = { 50, 80},
6503 <cmr>    \textlangle = {400, },
6504 <cmr>    \textrightangle = { , 400},
6505 <m-t|bch|pmn|ptm> \textminus = {200,200},
6506 <cmr|pad|tpl>    \textminus = {300,300},
6507 <blg|ugm>    \textminus = {250,300},
6508 <bch|pad|pmn> \textlbrackdbl = {100, },
6509 <blg>    \textlbrackdbl = {200, },
6510 <bch|pad|pmn> \textrbrackdbl = { , 100},
6511 <blg>    \textrbrackdbl = { , 200},
6512 <pmn>    \textasciigrave = {200,500},
6513 <bch|blg|cmr|pad|pmn> \texttildelow = {200,250},
6514 <pmn>    \textasciibreve = {300,400},
6515 <pmn>    \textasciicaron = {300,400},
6516 <pmn>    \textacutedbl = {200,300},
6517 <pmn>    \textgravedbl = {150,300},
6518 <bch|pmn|ugm> \textdagger = { 80, 80},
6519 <blg>    \textdagger = {200,200},
6520 <cmr|pad>    \textdagger = {100,100},
6521 <ptm>    \textdagger = {150,150},
6522 <blg>    \textdaggerdbl = {150,150},
6523 <cmr|pad|pmn> \textdaggerdbl = { 80, 80},
6524 <ptm>    \textdaggerdbl = {100,100},
6525 <bch>    \textbardbl = {100,100},
6526 <blg|ugm>    \textbardbl = {150,150},
6527 <bch>    \textbullet = {200,200},
6528 <blg>    \textbullet = {400,500},
6529 <cmr|pad|pmn> \textbullet = { , 100},
6530 <ptm>    \textbullet = {150,150},

```

```

6531 <ugm> \textbullet = { 50,100},
6532 <bch|cmr|pmn> \textcelsius = { 50, },
6533 <pad> \textcelsius = { 80, },
6534 <bch> \textflorin = { 50, 50},
6535 <blg> \textflorin = {100,100},
6536 <pad|ugm> \textflorin = { ,100},
6537 <pmn> \textflorin = { 50,100},
6538 <ptm> \textflorin = { 50, 70},
6539 <cmr> \textcolonmonetary = { , 50},
6540 <pad|pmn> \textcolonmonetary = { 50, },
6541 <pmn> \textinterrobang = { ,100},
6542 <pmn> \textinterrobangdown = {100, },
6543 <m-t|pad|ptm> \texttrademark = {100,100},
6544 <bch> \texttrademark = {150,150},
6545 <blg|cmr|ppl> \texttrademark = {200,200},
6546 <pmn> \texttrademark = { 50, 50},
6547 <ugm> \texttrademark = {100,150},
6548 <bch|ugm> \textcent = { 50, },
6549 <ptm> \textcent = {100,100},
6550 <bch> \textsterling = { 50, },
6551 <ugm> \textsterling = { , 50},
6552 <bch> \textbrokenbar = {200,200},
6553 <blg> \textbrokenbar = {250,250},
6554 <ugm> \textbrokenbar = {200,300},
6555 <pmn> \textasciidieresis = {300,400},
6556 <m-t|bch|cmr|pad|ptm|ugm> \textcopyright = {100,100},
6557 <pmn> \textcopyright = {100,150},
6558 <ppl> \textcopyright = {200,200},
6559 <bch|cmr|ugm> \textordfeminine = {100,200},
6560 <pad|pmn> \textordfeminine = {200,200},
6561 <bch|cmr|pad|pmn|ugm> \textlnot = {200, },
6562 <blg> \textlnot = {200,100},
6563 <m-t|bch|cmr|pad|ptm|ugm> \textregistered = {100,100},
6564 <pmn> \textregistered = { 50,150},
6565 <ppl> \textregistered = {200,200},
6566 <pmn> \textasciimacron = {150,200},
6567 <m-t|ppl|ptm> \textdegree = {300,300},
6568 <bch> \textdegree = {150,200},
6569 <blg|ugm> \textdegree = {200,200},
6570 <cmr|pad> \textdegree = {400,400},
6571 <pmn> \textdegree = {150,400},
6572 <bch|cmr|pad|pmn|ugm> \textpm = {150,200},
6573 <blg> \textpm = {100,100},
6574 <ptm> \textpm = { 50, 80},
6575 <bch|blg|ugm> \texttwosuperior = {100,200},
6576 <cmr> \texttwosuperior = { 50,100},
6577 <pad|pmn> \texttwosuperior = {200,200},
6578 <ptm> \texttwosuperior = { 50, 50},
6579 <bch|blg|ugm> \textthreesuperior = {100,200},
6580 <cmr> \textthreesuperior = { 50,100},
6581 <pad|pmn> \textthreesuperior = {200,200},
6582 <ptm> \textthreesuperior = { 50, 50},
6583 <pmn> \textasciiacute = {300,400},
6584 <bch|ugm> \textmu = { ,100},
6585 <bch|pad|pmn> \textparagraph = { ,100},
6586 <bch|cmr|pad|pmn> \textperiodcentered = {300,400},
6587 <blg> \textperiodcentered = {400,500},
6588 <ptm> \textperiodcentered = {300,300},
6589 <ugm> \textperiodcentered = {200,500},
6590 <bch|blg|ugm> \textonesuperior = {200,300},
6591 <cmr|pad|pmn> \textonesuperior = {200,200},
6592 <ptm> \textonesuperior = {100,100},
6593 <bch|pad|pmn|ugm> \textordmasculine = {200,200},
6594 <blg|cmr> \textordmasculine = {100,200},
6595 <bch|cmr|pmn> \texteuro = {100, },

```

```

6596 <pad> \texteuro = { 50,100},
6597 <bch> \texttimes = {200,200},
6598 <blg|ptm> \texttimes = {100,100},
6599 <cmr> \texttimes = {150,250},
6600 <pad> \texttimes = {100,150},
6601 <pmn> \texttimes = { 70,100},
6602 <ugm> \texttimes = {200,300},
6603 <bch|pad|pmn> \textdiv = {150,200}
6604 <blg> \textdiv = {100,100}
6605 <cmr> \textdiv = {150,250}
6606 <ptm> \textdiv = { 50,100},
6607 <ugm> \textdiv = {200,300},
6608 <ptm> \textperthousand = { ,50}
6609 <ugm> \textsection = { ,100},
6610 <ugm> \textonehalf = { 50,100},
6611 <ugm> \textonequarter = { 50,100},
6612 <ugm> \textthreequarters = { 50,100},
6613 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

6614 }
6615
6616 <*cmr|pad|pmn|ugm>
6617 \SetProtrusion
6618 <cmr> [ name = cmr-textcomp-it ]
6619 <pad> [ name = pad-textcomp-it ]
6620 <pmn> [ name = pmn-textcomp-it ]
6621 <ugm> [ name = ugm-textcomp-it ]
6622 { encoding = TSI,
6623 <cmr> family = cmr,
6624 <pad> family = {pad,padx,padj},
6625 <pmn> family = {pmnx,pmnj},
6626 <ugm> family = ugm,
6627 <!ugm> shape = {it,s1} }
6628 <ugm> shape = it }
6629 {
6630 <cmr> \textquotestraightbase = {300,600},
6631 <pad|pmn> \textquotestraightbase = {400,400},
6632 <cmr> \textquotestraightdblbase = {300,600},
6633 <pad> \textquotestraightdblbase = {300,400},
6634 <pmn> \textquotestraightdblbase = {300,300},
6635 \texttwelveudash = {200,200},
6636 <cmr|pad|pmn> \textthreequartersemdash = {150,150},
6637 <ugm> \textthreequartersemdash = {200,200},
6638 <cmr> \textquotesingle = {600,300},
6639 <pad> \textquotesingle = {800,100},
6640 <pmn> \textquotesingle = {300,200},
6641 <ugm> \textquotesingle = {500,500},
6642 <cmr> \textasteriskcentered = {300,200},
6643 <pad> \textasteriskcentered = {500,100},
6644 <pmn> \textasteriskcentered = {200,300},
6645 <ugm> \textasteriskcentered = {300,150},
6646 <pmn> \textfractionsolidus = {-200,-200},
6647 <cmr> \textoneoldstyle = {100, 50},
6648 <pad> \textoneoldstyle = {100, },
6649 <pmn> \textoneoldstyle = { 50, },
6650 <pad> \texttwooldstyle = { 50, },
6651 <pmn> \texttwooldstyle = {-50, },
6652 <cmr> \textthreeoldstyle = {100, 50},
6653 <pmn> \textthreeoldstyle = {-100, },
6654 <cmr> \textfouroldstyle = { 50, 50},
6655 <pad> \textfouroldstyle = { 50,100},
6656 <cmr> \textsevenoldstyle = { 50, 80},
6657 <pad> \textsevenoldstyle = { 50, },
6658 <pmn> \textsevenoldstyle = { 20, },

```

```

6659 <cmr> \textlangle = {400, },
6660 <cmr> \textrangle = { ,400},
6661 <cmr|pad> \textminus = {300,300},
6662 <pmn> \textminus = {200,200},
6663 <ugm> \textminus = {250,300},
6664 <pad|pmn> \textlbrackdbl = {100, },
6665 <pad|pmn> \textrbrackdbl = { ,100},
6666 <pmn> \textasciigrave = {300,300},
6667 <cmr|pad|pmn> \texttildelow = {200,250},
6668 <pmn> \textasciibreve = {300,300},
6669 <pmn> \textasciicaron = {300,300},
6670 <pmn> \textacutedbl = {200,300},
6671 <pmn> \textgravedbl = {150,300},
6672 <cmr> \textdagger = {100,100},
6673 <pad> \textdagger = {200,100},
6674 <pmn> \textdagger = { 80, 50},
6675 <ugm> \textdagger = { 80, 80},
6676 <cmr|pad> \textdaggerdbl = { 80, 80},
6677 <pmn> \textdaggerdbl = { 80, 50},
6678 <ugm> \textbardbl = {150,150},
6679 <cmr> \textbullet = {200,100},
6680 <pad> \textbullet = {300, },
6681 <pmn> \textbullet = { 30, 70},
6682 <ugm> \textbullet = { 50,100},
6683 <cmr> \textcelsius = {100, },
6684 <pad> \textcelsius = {200, },
6685 <pmn> \textcelsius = { 50,-50},
6686 <pad> \textflorin = {100, },
6687 <pmn> \textflorin = { 50,100},
6688 <ugm> \textflorin = { ,100},
6689 <cmr> \textcolonmonetary = {150, },
6690 <pad> \textcolonmonetary = {100, },
6691 <pmn> \textcolonmonetary = { 50,-50},
6692 <cmr|pad> \texttrademark = {200, },
6693 <pmn> \texttrademark = { 50,100},
6694 <ugm> \texttrademark = {150, 50},
6695 <ugm> \textcent = { 50, },
6696 <ugm> \textsterling = { , 50},
6697 <ugm> \textbrokenbar = {200,300},
6698 <pmn> \textasciidieresis = {300,200},
6699 <cmr> \textcopyright = {100, },
6700 <pad> \textcopyright = {200,100},
6701 <pmn> \textcopyright = {100,150},
6702 <ugm> \textcopyright = {300, },
6703 <cmr> \textordfeminine = {100,100},
6704 <pmn> \textordfeminine = {200,200},
6705 <ugm> \textordfeminine = {100,200},
6706 <cmr|pad> \textlnot = {300, },
6707 <pmn|ugm> \textlnot = {200, },
6708 <cmr> \textregistered = {100, },
6709 <pad> \textregistered = {200,100},
6710 <pmn> \textregistered = { 50,150},
6711 <ugm> \textregistered = {300, },
6712 <pmn> \textasciimacron = {150,200},
6713 <cmr|pad> \textdegree = {500,100},
6714 <pmn> \textdegree = {150,150},
6715 <ugm> \textdegree = {300,200},
6716 <cmr> \textpm = {150,100},
6717 <pad> \textpm = {200,150},
6718 <pmn|ugm> \textpm = {150,200},
6719 <cmr> \textonesuperior = {400, },
6720 <pad> \textonesuperior = {300,100},
6721 <pmn> \textonesuperior = {200,100},
6722 <ugm> \textonesuperior = {300,300},
6723 <cmr> \texttwosuperior = {400, },

```

```

6724 <pad> \texttwosuperior = {300, },
6725 <pmn> \texttwosuperior = {200,100},
6726 <ugm> \texttwosuperior = {300,200},
6727 <cmr> \textthreesuperior = {400, },
6728 <pad> \textthreesuperior = {300, },
6729 <pmn> \textthreesuperior = {200,100},
6730 <ugm> \textthreesuperior = {300,200},
6731 <ugm> \textmu = { ,100},
6732 <pmn> \textasciicute = {300,200},
6733 <cmr> \textparagraph = {200, },
6734 <pmn> \textparagraph = { ,100},
6735 <cmr> \textperiodcentered = {500,500},
6736 <pad|pmn|ugm> \textperiodcentered = {300,400},
6737 <cmr> \textordmasculine = {100,100},
6738 <pmn> \textordmasculine = {200,200},
6739 <ugm> \textordmasculine = {300,200},
6740 <cmr> \texteuro = {200, },
6741 <pad> \texteuro = {100, },
6742 <pmn> \texteuro = {100,-50},
6743 <cmr> \texttimes = {200,200},
6744 <pad> \texttimes = {200,100},
6745 <pmn> \texttimes = { 70,100},
6746 <ugm> \texttimes = {200,300},
6747 <cmr|pad> \textdiv = {200,200}
6748 <pmn> \textdiv = {150,200}
6749 <ugm> \textdiv = {200,300},
6750 <ugm> \textsection = { ,200},
6751 <ugm> \textonehalf = { 50,100},
6752 <ugm> \textonequarter = { 50,100},
6753 <ugm> \textthreequarters = { 50,100},
6754 <ugm> \textsurd = { ,100}
6755 }
6756
6757 </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}

```

```

6758 <*cmr>
6759 \SetProtrusion
6760 [ name = cmr-math-letters ]
6761 { encoding = OML,
6762   family = cmm,
6763   series = {m,b},
6764   shape = it }
6765 {
6766   A = {100, 50}, % \mathnormal
6767   B = { 50, },

```

```

6768     C = { 50,  },
6769     D = { 50, 50},
6770     E = { 50,  },
6771     F = {100, 50},
6772     G = { 50, 50},
6773     H = { 50, 50},
6774     I = { 50, 50},
6775     J = {150, 50},
6776     K = { 50,100},
6777     L = { 50, 50},
6778     M = { 50,  },
6779     N = { 50,  },
6780     O = { 50,  },
6781     P = { 50,  },
6782     Q = { 50, 50},
6783     R = { 50,  },
6784     S = { 50,  },
6785     T = { 50,100},
6786     U = { 50, 50},
6787     V = {100,100},
6788     W = { 50,100},
6789     X = { 50,100},
6790     Y = {100,100},
6791     f = {100,100},
6792     h = {  ,100},
6793     i = {  , 50},
6794     j = {  , 50},
6795     k = {  , 50},
6796     r = {  , 50},
6797     v = {  , 50},
6798     w = {  , 50},
6799     x = {  , 50},
6800     "0B = { 50,100}, % \alpha
6801     "0C = { 50, 50}, % \beta
6802     "0D = {200,150}, % \gamma
6803     "0E = { 50, 50}, % \delta
6804     "0F = { 50, 50}, % \epsilon
6805     "10 = { 50,150}, % \zeta
6806     "12 = { 50,  }, % \theta
6807     "13 = {  ,100}, % \iota
6808     "14 = {  ,100}, % \kappa
6809     "15 = {100, 50}, % \lambda
6810     "16 = {  , 50}, % \mu
6811     "17 = {  , 50}, % \nu
6812     "18 = {  , 50}, % \xi
6813     "19 = { 50,100}, % \pi
6814     "1A = { 50, 50}, % \rho
6815     "1B = {  ,150}, % \sigma
6816     "1C = { 50,150}, % \tau
6817     "1D = { 50, 50}, % \upsilon
6818     "1F = { 50,100}, % \chi
6819     "20 = { 50, 50}, % \psi
6820     "21 = {  , 50}, % \omega
6821     "22 = {  , 50}, % \varepsilon
6822     "23 = {  , 50}, % \vartheta
6823     "24 = {  , 50}, % \varpi
6824     "25 = {100,  }, % \varrho
6825     "26 = {100,100}, % \varsigma
6826     "27 = { 50, 50}, % \varphi
6827     "28 = {100,100}, % \leftharpoonup
6828     "29 = {100,100}, % \leftharpoondown
6829     "2A = {100,100}, % \rightharpoonup
6830     "2B = {100,100}, % \rightharpoondown
6831     "2C = {300,200}, % \lhook
6832     "2D = {200,300}, % \rhook

```

```

6833 "2E = { ,100}, % \triangleright
6834 "2F = {100, }, % \triangleleft
6835 "3A = { ,500}, % ., \ldotp
6836 "3B = { ,500}, % ,
6837 "3C = {200,100}, % <
6838 "3D = {300,400}, % /
6839 "3E = {100,200}, % >
6840 "3F = {200,200}, % \star
6841 "5B = { ,100}, % \flat
6842 "5E = {200,200}, % \smile
6843 "5F = {200,200}, % \frown
6844 "7C = {100, }, % \jmath
6845 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

6846 }
6847

```

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}

```

```

6848 \SetProtrusion
6849 [ name = cmr-math-symbols ]
6850 { encoding = OMS,
6851   family = cmsy,
6852   series = {m,b},
6853   shape = n }
6854 {
6855   A = {150, 50}, % \mathcal
6856   C = { ,100},
6857   D = { , 50},
6858   F = { 50,150},
6859   I = { ,100},
6860   J = {100,150},
6861   K = { ,100},
6862   L = {100, },
6863   M = { 50, 50},
6864   N = { 50,100},
6865   P = { , 50},
6866   Q = { 50, },
6867   R = { , 50},
6868   T = { 50,150},
6869   V = { 50, 50},
6870   W = { , 50},
6871   X = {100,100},
6872   Y = {100, },
6873   Z = {100,150},
6874 "00 = {300,300}, % -
6875 "01 = { ,700}, % \cdot, \cdotp
6876 "02 = {150,250}, % \times
6877 "03 = {150,250}, % *, \ast
6878 "04 = {200,300}, % \div
6879 "05 = {150,250}, % \diamond
6880 "06 = {200,200}, % \pm
6881 "07 = {200,200}, % \mp
6882 "08 = {100,100}, % \oplus
6883 "09 = {100,100}, % \ominus
6884 "0A = {100,100}, % \otimes
6885 "0B = {100,100}, % \oslash
6886 "0C = {100,100}, % \odot
6887 "0D = {100,100}, % \bigcirc
6888 "0E = {100,100}, % \circ
6889 "0F = {100,100}, % \bullet
6890 "10 = {100,100}, % \asymp

```

```

6891 "11 = {100,100}, % \equiv
6892 "12 = {200,100}, % \subseteq
6893 "13 = {100,200}, % \supseteq
6894 "14 = {200,100}, % \leq
6895 "15 = {100,200}, % \geq
6896 "16 = {200,100}, % \preceq
6897 "17 = {100,200}, % \succeq
6898 "18 = {200,200}, % \sim
6899 "19 = {150,150}, % \approx
6900 "1A = {200,100}, % \subset
6901 "1B = {100,200}, % \supset
6902 "1C = {200,100}, % \ll
6903 "1D = {100,200}, % \gg
6904 "1E = {300,100}, % \prec
6905 "1F = {100,300}, % \succ
6906 "20 = {100,200}, % \leftarrow
6907 "21 = {200,100}, % \rightarrow
6908 "22 = {100,100}, % \uparrow
6909 "23 = {100,100}, % \downarrow
6910 "24 = {100,100}, % \leftrightarrows
6911 "25 = {100,100}, % \nearrow
6912 "26 = {100,100}, % \searrow
6913 "27 = {100,100}, % \simeq
6914 "28 = {100,100}, % \Leftarrow
6915 "29 = {100,100}, % \Rightarrow
6916 "2A = {100,100}, % \Uparrow
6917 "2B = {100,100}, % \Downarrow
6918 "2C = {100,100}, % \Leftrightarrow
6919 "2D = {100,100}, % \nwarrow
6920 "2E = {100,100}, % \swarrow
6921 "2F = { ,100}, % \propto
6922 "30 = { ,400}, % \prime
6923 "31 = {100,100}, % \infty
6924 "32 = {150,100}, % \in
6925 "33 = {100,150}, % \ni
6926 "34 = {100,100}, % \triangle, \bigtriangleup
6927 "35 = {100,100}, % \bigtriangledown
6928 "38 = { ,100}, % \forall
6929 "39 = {100, }, % \exists
6930 "3A = {200, }, % \neg
6931 "3E = {200,200}, % \top
6932 "3F = {200,200}, % \bot, \perp
6933 "5E = {100,200}, % \wedge
6934 "5F = {100,200}, % \vee
6935 "60 = { ,300}, % \vdash
6936 "61 = {300, }, % \dashv
6937 "62 = {100,100}, % \lfloor
6938 "63 = {100,100}, % \rfloor
6939 "64 = {100,100}, % \lceil
6940 "65 = {100,100}, % \rceil
6941 "66 = {150, }, % \lbrace
6942 "67 = { ,150}, % \rbrace
6943 "68 = {400, }, % \langle
6944 "69 = { ,400}, % \rangle
6945 "6C = {100,100}, % \updownarrow
6946 "6D = {100,100}, % \Updownarrow
6947 "6E = {100,300}, % \, \backslash, \setminus
6948 "72 = {100,100}, % \nabla
6949 "79 = {200,200}, % \dagger
6950 "7A = {100,100}, % \ddagger
6951 "7B = {100, }, % \mathparagraph
6952 "7C = {100,100}, % \clubsuit
6953 "7D = {100,100}, % \diamondsuit
6954 "7E = {100,100}, % \heartsuit
6955 "7F = {100,100} % \spadesuit

```


Remaining slots in the source file.

```
6956 }
6957
```

We don't bother about 'largesymbols', since it will only be used in display math, where protrusion doesn't work anyway. It's declared as:

```
\DeclareSymbolFont{largesymbols}{OMX}{cmex}{m}{n}
```

```
6958 </cmr>
6959 </cfg-t>
```

15.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
6960 <*cfg-u>
```

Symbol font 'a'.

```
6961 <*msa>
6962 \SetProtrusion
6963 [ name = AMS-a ]
6964 { encoding = U,
6965   family = msa }
6966 {
6967   "05 = {150,250}, % \centerdot
6968   "06 = {100,100}, % \lozenge
6969   "07 = { 50, 50}, % \blacklozenge
6970   "08 = { 50, 50}, % \circlearrowright
6971   "09 = { 50, 50}, % \circlearrowleft
6972   "0A = {100,100}, % \rightleftharpoons
6973   "0B = {100,100}, % \leftrightharpoons
6974   "0D = {-50,200}, % \Vdash
6975   "0E = {-50,200}, % \Vvdash
6976   "0F = {-70,150}, % \vDash
6977   "10 = {100,150}, % \twoheadrightarrow
6978   "11 = {100,150}, % \twoheadleftarrow
6979   "12 = { 50,100}, % \leftleftarrows
6980   "13 = { 50, 80}, % \rightrightarrows
6981   "14 = {120,120}, % \upuparrows
6982   "15 = {120,120}, % \downdownarrows
6983   "16 = {200,200}, % \upharpoonright
6984   "17 = {200,200}, % \downharpoonright
6985   "18 = {200,200}, % \upharpoonleft
6986   "19 = {200,200}, % \downharpoonleft
6987   "1A = { 80,100}, % \rightarrowtail
6988   "1B = { 80,100}, % \leftarrowtail
6989   "1C = { 50, 50}, % \leftrightarrows
6990   "1D = { 50, 50}, % \rightleftarrows
6991   "1E = {250, }, % \Lsh
6992   "1F = { ,250}, % \Rsh
6993   "20 = {100,100}, % \rightsquigarrow
6994   "21 = {100,100}, % \leftrightsquigarrow
6995   "22 = {100, 50}, % \looparrowleft
6996   "23 = { 50,100}, % \looparrowright
6997   "24 = { 50, 80}, % \circeq
6998   "25 = { ,100}, % \succsim
6999   "26 = { ,100}, % \gtrsim
7000   "27 = { ,100}, % \gtrapprox
7001   "28 = {150, 50}, % \multimap
7002   "2B = {100,150}, % \doteqdot
7003   "2C = {100,150}, % \triangleq
7004   "2D = {100, 50}, % \precsim
7005   "2E = {100, 50}, % \lesssim
7006   "2F = { 50, 50}, % \lessapprox
```

```

7007 "30 = {100, 50}, % \eqslantless
7008 "31 = { 50, 50}, % \eqslantgtr
7009 "32 = {100, 50}, % \curlyeqprec
7010 "33 = { 50,100}, % \curlyeqsucc
7011 "34 = {100, 50}, % \preccurlyeq
7012 "36 = { 50,  }, % \leqslant
7013 "38 = {  , 50}, % \backprime
7014 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
7015 "3C = { 50,100}, % \succcurlyeq
7016 "3E = {  , 50}, % \geqslant
7017 "40 = {  , 50}, % \sqsubset
7018 "41 = { 50,  }, % \sqsupset
7019 "42 = {  ,150}, % \vartriangleright, \rhd
7020 "43 = {150,  }, % \vartriangleleft, \lhd
7021 "44 = {  ,100}, % \trianglerighteq, \unrhd
7022 "45 = {100,  }, % \trianglelefteq, \unlhd
7023 "46 = {100,100}, % \bigstar
7024 "48 = { 50, 50}, % \blacktriangledown
7025 "49 = {  ,100}, % \blacktriangleright
7026 "4A = {100,  }, % \blacktriangleleft
7027 "4B = {  ,150}, % \dashrightarrow (the arrow)
7028 "4C = {150,  }, % \dashleftarrow
7029 "4D = { 50, 50}, % \vartriangle
7030 "4E = { 50, 50}, % \blacktriangle
7031 "4F = { 50, 50}, % \triangledown
7032 "50 = { 50, 50}, % \eqcirc
7033 "56 = {  ,150}, % \Rrightarrow
7034 "57 = {150,  }, % \Lleftarrow
7035 "58 = {100,300}, % \checkmark
7036 "5C = { 50, 50}, % \angle
7037 "5D = { 50, 50}, % \measuredangle
7038 "5E = { 50, 50}, % \sphericalangle
7039 "5F = {  , 50}, % \varpropto
7040 "60 = {100,100}, % \smallsmile
7041 "61 = {100,100}, % \smallfrown
7042 "62 = { 50,  }, % \Subset
7043 "63 = {  , 50}, % \Supset
7044 "66 = {150,150}, % \curlywedge
7045 "67 = {150,150}, % \curlyvee
7046 "68 = { 50,150}, % \leftthreetimes
7047 "69 = {100, 50}, % \rightthreetimes
7048 "6C = { 50, 50}, % \bumpeq
7049 "6D = { 50, 50}, % \Bumpeq
7050 "6E = {100,  }, % \lll
7051 "6F = {  ,100}, % \ggg
7052 "70 = { 50,100}, % \ulcorner
7053 "71 = {100, 50}, % \urcorner
7054 "75 = {150,200}, % \dotplus
7055 "76 = { 50,100}, % \backsimeq
7056 "78 = { 50,100}, % \llcorner
7057 "79 = {100, 50}, % \lrcorner
7058 "7C = {100,100}, % \intercal
7059 "7D = { 50, 50}, % \circledcirc
7060 "7E = { 50, 50}, % \circledast
7061 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

7062 }
7063
7064 </msa>

```

Symbol font 'b'.

```

7065 <*msb>
7066 \SetProtrusion
7067 [ name = AMS-b ]

```

```

7068 { encoding = U,
7069   family   = msb }
7070 {
7071   A = { 50, 50}, % \mathbb
7072   C = { 50, 50},
7073   G = {   , 50},
7074   L = {   , 50},
7075   P = {   , 50},
7076   R = {   , 50},
7077   T = {   , 50},
7078   V = { 50, 50},
7079   X = { 50, 50},
7080   Y = { 50, 50},
7081 "00 = { 50, 50}, % \lvertneqq
7082 "01 = { 50, 50}, % \gvertneqq
7083 "02 = { 50, 50}, % \nleq
7084 "03 = { 50, 50}, % \ngeq
7085 "04 = {100, 50}, % \nless
7086 "05 = { 50,150}, % \ngtr
7087 "06 = {100, 50}, % \nprec
7088 "07 = { 50,150}, % \nsucc
7089 "08 = { 50, 50}, % \lneqq
7090 "09 = { 50, 50}, % \gneqq
7091 "0A = {100,100}, % \nleqslant
7092 "0B = {100,100}, % \ngeqslant
7093 "0C = {100, 50}, % \lneq
7094 "0D = { 50,100}, % \gneq
7095 "0E = {100, 50}, % \npreceq
7096 "0F = { 50,100}, % \nsucceq
7097 "10 = { 50,   }, % \precnsim
7098 "11 = { 50, 50}, % \succnsim
7099 "12 = { 50, 50}, % \lnsim
7100 "13 = { 50, 50}, % \gnsim
7101 "14 = { 50, 50}, % \lneqq
7102 "15 = { 50, 50}, % \ngeqq
7103 "16 = { 50, 50}, % \precneqq
7104 "17 = { 50, 50}, % \succneqq
7105 "18 = { 50, 50}, % \precnapprox
7106 "19 = { 50, 50}, % \succnapprox
7107 "1A = { 50, 50}, % \lnapprox
7108 "1B = { 50, 50}, % \gnapprox
7109 "1C = {150,200}, % \nsim
7110 "1D = { 50, 50}, % \ncong
7111 "1E = {100,150}, % \diagup
7112 "1F = {100,150}, % \diagdown
7113 "20 = {100, 50}, % \varsubsetneq
7114 "21 = { 50,100}, % \varsupsetneq
7115 "22 = {100, 50}, % \subsetneqq
7116 "23 = { 50,100}, % \supsetneqq
7117 "24 = {100, 50}, % \subsetneqq
7118 "25 = { 50,100}, % \supsetneqq
7119 "26 = {100, 50}, % \varsubsetneqq
7120 "27 = { 50,100}, % \varsupsetneqq
7121 "28 = {100, 50}, % \subsetneq
7122 "29 = { 50,100}, % \supsetneq
7123 "2A = {100, 50}, % \subteq
7124 "2B = { 50,100}, % \supseteq
7125 "2C = { 50,100}, % \nparallel
7126 "2D = {100,150}, % \nmid
7127 "2E = {150,150}, % \nshortmid
7128 "2F = {100,100}, % \nshortparallel
7129 "30 = {   ,150}, % \nvdash
7130 "31 = {   ,150}, % \nVDash
7131 "32 = {   ,100}, % \nvDash
7132 "33 = {   ,100}, % \nVDash

```

```

7133 "34 = { ,100}, % \ntrianglerighteq
7134 "35 = {100, }, % \ntrianglelefteq
7135 "36 = {100, }, % \ntriangleleft
7136 "37 = { ,100}, % \ntriangleright
7137 "38 = {100,200}, % \nleftarrow
7138 "39 = {100,200}, % \nrightrightarrow
7139 "3A = {100,100}, % \nLeftarrow
7140 "3B = { 50,100}, % \nRrightarrow
7141 "3C = {100,100}, % \nLeftrightarrow
7142 "3D = {100,200}, % \nleftrightharrow
7143 "3E = { 50, 50}, % \divideontimes
7144 "3F = { 50, 50}, % \varnothing
7145 "60 = {200, }, % \Finv
7146 "61 = { , 50}, % \Game
7147 "68 = {100,100}, % \eqsim
7148 "69 = { 50, }, % \beth
7149 "6A = { 50, }, % \gimel
7150 "6B = {150, }, % \daleth
7151 "6C = {200, }, % \lessdot
7152 "6D = { ,200}, % \gtrdot
7153 "6E = {100,200}, % \ltimes
7154 "6F = {150,100}, % \rtimes
7155 "70 = { 50,100}, % \shortmid
7156 "71 = { 50, 50}, % \shortparallel
7157 "72 = {200,300}, % \smallsetminus
7158 "73 = {100,200}, % \thicksim
7159 "74 = { 50,100}, % \thickapprox
7160 "75 = { 50, 50}, % \approx
7161 "76 = { 50,100}, % \succapprox
7162 "77 = { 50, 50}, % \precapprox
7163 "78 = {100,100}, % \curvearrowleft
7164 "79 = { 50,150}, % \curvearrowright
7165 "7A = { 50,200}, % \digamma
7166 "7B = {100, 50}, % \varkappa
7167 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

7168 }
7169
7170 (/msb)

```

15.8.8 Euler

Euler Roman font (package eulr).

```

7171 (*eur)
7172 \SetProtrusion
7173 [ name = eulr ]
7174 { encoding = U,
7175 family = eur }
7176 {
7177 "01 = {100,100},
7178 "03 = {100,150},
7179 "06 = { ,100},
7180 "07 = {100,150},
7181 "08 = {100,100},
7182 "0A = {100,100},
7183 "0B = { , 50},
7184 "0C = { ,100},
7185 "0D = {100,100},
7186 "0E = { ,100},
7187 "0F = {100,100},
7188 "10 = {100,100},
7189 "13 = { ,100},
7190 "14 = { ,100},

```

```

7191     "15 = {   , 50},
7192     "16 = {   , 50},
7193     "17 = { 50,100},
7194     "18 = { 50,100},
7195     "1A = {   , 50},
7196     "1B = {   , 50},
7197     "1C = { 50,100},
7198     "1D = { 50,100},
7199     "1E = { 50,100},
7200     "1F = { 50,100},
7201     "20 = {   , 50},
7202     "21 = {   , 50},
7203     "22 = { 50,100},
7204     "24 = {   , 50},
7205     "27 = { 50,100},
7206     1  = {100,100},
7207     7  = { 50,100},
7208     "3A = {300,500},
7209     "3B = {200,400},
7210     "3C = {200,100},
7211     "3D = {200,200},
7212     "3E = {100,200},
7213     A  = {   ,100},
7214     D  = {   , 50},
7215     J  = { 50,   },
7216     K  = {   , 50},
7217     L  = {   , 50},
7218     Q  = {   , 50},
7219     T  = { 50,   },
7220     X  = { 50, 50},
7221     Y  = { 50,   },
7222     h  = {   , 50},
7223     k  = {   , 50}
7224 }
7225

```

Extended by the `eulervm` package.

```

7226 \SetProtrusion
7227 [ name      = euler-vm,
7228   load      = euler ]
7229 { encoding = U,
7230   family   = zeur }
7231 {
7232   "28 = {100,200},
7233   "29 = {100,200},
7234   "2A = {100,150},
7235   "2B = {100,150},
7236   "2C = {200,300},
7237   "2D = {200,300},
7238   "2E = {   ,100},
7239   "2F = {100,   },
7240   "3F = {150,150},
7241   "5B = {   ,100},
7242   "5E = {100,100},
7243   "5F = {100,100},
7244   "80 = {   , 50},
7245   "81 = {200,250},
7246   "82 = {100,200}
7247 }
7248
7249 </eur>

```

Euler Script font (`euca1`).

```

7250 < *eus >
7251 \SetProtrusion

```

```
7252 [ name = euscript ]
7253 { encoding = U,
7254   family = eus }
7255 {
7256   A = {100,100},
7257   B = { 50,100},
7258   C = { 50, 50},
7259   D = { 50,100},
7260   E = { 50,100},
7261   F = { 50,  },
7262   G = { 50,  },
7263   H = {  ,100},
7264   K = {  , 50},
7265   L = {  ,150},
7266   M = {  , 50},
7267   N = {  , 50},
7268   O = { 50, 50},
7269   P = { 50, 50},
7270   T = {  ,100},
7271   U = {  , 50},
7272   V = { 50, 50},
7273   W = { 50, 50},
7274   X = { 50, 50},
7275   Y = { 50,  },
7276   Z = { 50,100},
7277   "00 = {250,250},
7278   "18 = {200,200},
7279   "3A = {200,150},
7280   "40 = {  ,100},
7281   "5E = {100,100},
7282   "5F = {100,100},
7283   "66 = { 50,  },
7284   "67 = {  , 50},
7285   "6E = {200,200}
7286 }
7287
7288 \SetProtrusion
7289 [ name = euscript-vm,
7290   load = euscript ]
7291 { encoding = U,
7292   family = zeus }
7293 {
7294   "01 = {600,600},
7295   "02 = {200,200},
7296   "03 = {200,200},
7297   "04 = {200,200},
7298   "05 = {150,150},
7299   "06 = {200,200},
7300   "07 = {200,200},
7301   "08 = {100,100},
7302   "09 = {100,100},
7303   "0A = {100,100},
7304   "0B = {100,100},
7305   "0C = {100,100},
7306   "0D = {100,100},
7307   "0E = {150,150},
7308   "0F = {100,100},
7309   "10 = {150,150},
7310   "11 = {100,100},
7311   "12 = {150,100},
7312   "13 = {100,150},
7313   "14 = {150,100},
7314   "15 = {100,150},
7315   "16 = {200,100},
7316   "17 = {100,200},
```

```

7317     "19 = {150,150},
7318     "1A = {150,100},
7319     "1B = {100,150},
7320     "1C = {100,100},
7321     "1D = {100,100},
7322     "1E = {250,100},
7323     "1F = {100,250},
7324     "20 = {150,200},
7325     "21 = {150,200},
7326     "22 = {150,150},
7327     "23 = {150,150},
7328     "24 = {100,200},
7329     "25 = {150,150},
7330     "26 = {150,150},
7331     "27 = {100,100},
7332     "28 = {100,100},
7333     "29 = {100,150},
7334     "2A = {100,100},
7335     "2B = {100,100},
7336     "2C = {100,100},
7337     "2D = {150,150},
7338     "2E = {150,150},
7339     "2F = {100,100},
7340     "30 = {100,100},
7341     "31 = {100,100},
7342     "32 = {100,100},
7343     "33 = {100,100},
7344     "34 = {100,100},
7345     "35 = {100,100},
7346     "3E = {150,150},
7347     "3F = {150,150},
7348     "60 = {   ,200},
7349     "61 = {200,   },
7350     "62 = {100,100},
7351     "63 = {100,100},
7352     "64 = {100,100},
7353     "65 = {100,100},
7354     "68 = {300,   },
7355     "69 = {   ,300},
7356     "6C = {100,100},
7357     "6D = {100,100},
7358     "6F = {100,100},
7359     "72 = {100,100},
7360     "73 = {200,100},
7361     "76 = {   ,100},
7362     "77 = {100,   },
7363     "78 = { 50, 50},
7364     "79 = {100,100},
7365     "7A = {100,100},
7366     "7D = {150,150},
7367     "7E = {100,100},
7368     "A8 = {100,100},
7369     "A9 = {100,100},
7370     "AB = {200,200},
7371     "BA = {   ,200},
7372     "BB = {   ,200},
7373     "BD = {200,200},
7374     "DE = {200,200}
7375   }
7376
7377 </eus>

Euler Fraktur font (eufrak).
7378 <+euf>
7379 \SetProtrusion

```

```

7380 [ name = mathfrak ]
7381 { encoding = U,
7382   family = euf }
7383 {
7384   A = { , 50},
7385   B = { , 50},
7386   C = { 50, 50},
7387   D = { , 80},
7388   E = { 50, },
7389   G = { , 50},
7390   L = { , 80},
7391   O = { , 50},
7392   T = { , 80},
7393   X = { 80, 50},
7394   Z = { 80, 50},
7395   b = { , 50},
7396   c = { , 50},
7397   k = { , 50},
7398   p = { , 50},
7399   q = { 50, },
7400   v = { , 50},
7401   w = { , 50},
7402   x = { , 50},
7403   1 = {100,100},
7404   2 = { 80, 80},
7405   3 = { 80, 50},
7406   4 = { 80, 50},
7407   7 = { 50, 50},
7408   "12 = {500,500},
7409   "13 = {500,500},
7410   ! = { ,200},
7411   ' = {200,300},
7412   ( = {200, },
7413   ) = { ,200},
7414   * = {200,200},
7415   + = {200,250},
7416   - = {200,200},
7417   {,} = {300,300},
7418   . = {400,400},
7419   {=} = {200,200},
7420   : = { ,200},
7421   ; = { ,200},
7422   ] = { ,200}
7423 }
7424
7425 </euf>
7426 </cfg-u>

```

15.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym²⁴).

```

7427 <*cfg-e>
7428 \SetProtrusion
7429 <zpeu|euroitc> { encoding = U,
7430   <mvs> { encoding = {OT1,U},
7431   <zpeu> family = zpeu }
7432   <euroitc> family = {euroitc,euroitcs} }
7433   <mvs> family = mvs }
7434   {
7435   <zpeu> E = {50, }
7436   <euroitc> E = {100,50}

```

24 Of course, there are many more symbols in this font. Feel free to contribute protrusion settings!

Figure 1:

Example for interword spacing (from Siemoneit 1989). The numbers indicate the preference/order when the interword space needs to be shrunk.

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

```

7437 <mvs>      164 = {50,50},    % \EUR
7438 <mvs>      068 = {50,-100} % \EURdig
7439 }
7440
7441 <*zpeu|euroitc>
7442 \SetProtrusion
7443 { encoding = U,
7444 <zpeu>      family = zpeu,
7445 <euroitc>   family = {euroitc,euroitcs},
7446   shape    = it* }
7447 {
7448 <zpeu>      E = {100,-50}
7449 <euroitc>   E = {100,}
7450 }
7451
7452 </zpeu|euroitc>
7453 <*zpeu>
7454 \SetProtrusion
7455 { encoding = U,
7456   family = {zpeus,eurosans} }
7457 {
7458   E = {100,50}
7459 }
7460
7461 \SetProtrusion
7462 { encoding = U,
7463   family = {zpeus,eurosans},
7464   shape  = it* }
7465 {
7466   E = {200, }
7467 }
7468
7469 </zpeu>
7470 </cfg-e>

```

15.9 Interword spacing

Default unit is space.

```

7471 <*m-t|cmr>
7472 %%% -----
7473 %%% INTERWORD SPACING
7474 </m-t|cmr>
7475
7476 <*m-t>
7477 \SetExtraSpacing
7478 [ name = default ]
7479 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
7480 {

```

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

7481 { , } = { , -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)

7482 r = { , -300, 300 } ,

- [before or] after lowercase characters with ascenders

7483 b = { , -200, 200 } ,

7484 d = { , -200, 200 } ,

7485 f = { , -200, 200 } ,

7486 h = { , -200, 200 } ,

7487 k = { , -200, 200 } ,

7488 l = { , -200, 200 } ,

7489 t = { , -200, 200 } ,

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’

7490 c = { , -100, 100 } ,

7491 p = { , -100, 100 } ,

7492 v = { , -100, 100 } ,

7493 w = { , -100, 100 } ,

7494 z = { , -100, 100 } ,

7495 x = { , -100, 100 } ,

7496 y = { , -100, 100 } ,

- [before or] after lowercase characters with x-height plus descender without additional optical space

7497 i = { , 50, -50 } ,

7498 m = { , 50, -50 } ,

7499 n = { , 50, -50 } ,

7500 u = { , 50, -50 } ,

- after colon and semicolon

7501 : = { , 200, -200 } ,

7502 ; = { , 200, -200 } ,

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

7503 . = { , 250, -250 } ,

7504 ! = { , 250, -250 } ,

7505 ? = { , 250, -250 } ,

The order has to be reversed when enlarging is needed.’

7506 }

7507

7508 </m-t>

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000 .)
- Should the first parameter also be used? (Probably.)
- What about quotation marks, parentheses etc.?

Furthermore, there seems to be a pdfTeX bug with spacing in combination with a non-zero `\spaceskip` (reported by *Axel Berger*):

```
\parfillskip0pt
\rightskip0pt plus 1em
\spaceskip\fontdimen2\font
test test\par
\pdfadjustinterwordglue2
\stbscode\font`t=-50
test test
\bye
```

Some more characters in T2A.²⁵

```
7509 <*cmr>
7510 \SetExtraSpacing
7511   [ name      = T2A,
7512     load      = default ]
7513   { encoding = T2A,
7514     family   = cmr }
7515   {
7516     \cyrg = { , -300, 300},
7517     \cyrb = { , -200, 200},
7518     \cyrk = { , -200, 200},
7519     \cyrs = { , -100, 100},
7520     \cyrr = { , -100, 100},
7521     \cyrh = { , -100, 100},
7522     \cyru = { , -100, 100},
7523     \cyrt = { , 50, -50},
7524     \cyrp = { , 50, -50},
7525     \cyri = { , 50, -50},
7526     \cyrishrt = { , 50, -50},
7527   }
7528
```

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’ (`\fontdimen7`) for Computer Modern Roman is a third of `\fontdimen2`, i.e., 333.

```
7529 \SetExtraSpacing
7530   [ name      = nonfrench-cmr,
7531     load      = default,
7532     context   = nonfrench ]
7533   { encoding = {OT1,T1,LY1,OT4,QX,T5},
```

25 Contributed by *Karl Karlsson*.

```
7534     family = cmr }
7535     {
```

latex.ltx has:

```
\def\nonfrenchspacing{
  \sfcode`. 3000
  \sfcode`? 3000
  \sfcode`! 3000
```

```
7536     . = {333,2000,-667},
7537     ? = {333,2000,-667},
7538     ! = {333,2000,-667},
```

```
\sfcode`: 2000
```

```
7539     : = {333,1000,-500},
```

```
\sfcode`; 1500
```

```
7540     ; = {   , 500,-333},
```

```
\sfcode`\, 1250
```

```
7541     {,}= {   , 250,-200}
```

```
}
```

```
7542   }
7543
7544 </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.

```
7545 <*-t>
7546 \SetExtraSpacing
7547 [ name = nonfrench-default,
7548   load = default,
7549   context = nonfrench ]
7550 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
7551 {
7552   . = {240,2000,-667},
7553   ? = {240,2000,-667},
7554   ! = {240,2000,-667},
7555   : = {240,1000,-500},
7556   ; = {   , 500,-333},
7557   {,}= {   , 250,-200}
7558 }
7559
```

15.10 Additional kerning

Default unit is 1 em.

```
7560 %%% -----
7561 %%% ADDITIONAL KERNING
7562
```

A dummy list to be loaded when no context is active.

```
7563 \SetExtraKerning
7564 [ name = empty ]
7565 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
7566 { }
7567
```

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. (Wikipedia²⁶ 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.)

```
7568 \SetExtraKerning
7569   [ name      = french-default,
7570     context   = french,
7571     unit      = space ]
7572   { encoding = {OT1,T1,LY1} }
7573   {
7574     : = {1000,}, % = \fontdimen2
7575     ; = {500,}, % = \thinspace
7576     ! = {500,},
7577     ? = {500,}
7578   }
7579
```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```
7580 \SetExtraKerning
7581   [ name      = french-guillemets,
7582     context   = french-guillemets,
7583     load      = french-default,
7584     unit      = space ]
7585   { encoding = {T1,LY1} }
7586   {
7587     \guillemotleft = { ,800}, % = 0.8\fontdimen2
7588     \guillemotright = {800,}
7589   }
7590
7591 \SetExtraKerning
7592   [ name      = french-guillemets-OT1,
7593     context   = french-guillemets,
7594     load      = french-default,
7595     unit      = space ]
7596   { encoding = OT1 }
7597   { }
7598
```

15.10.2 Turkish

```
7599 \SetExtraKerning
7600   [ name      = turkish,
7601     context   = turkish ]
7602   { encoding = {OT1,T1,LY1} }
7603   {
7604     : = {167,}, % = \thinspace
7605     ! = {167,},
7606     {=} = {167,}
7607   }
7608
7609 </m-t>
7610 </config>
```

26 http://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

7697 I,İ,I,I}, % Cyr
 7698 J = {Ĵ,
 7699 J}, % Cyr
 7700 K = {Რ,Ს,Ტ,Უ,Ფ,
 7701 K,Ქ,Ღ,Ყ,Შ,Ჩ,Ც}, % Cyr
 7702 L = {Წ,Ჭ,Ხ,Ჯ,Ჰ,Ჱ,Ჲ}, % L
 7703 M = {Ჴ,Ჵ,Ჶ,
 7704 M,Ჸ}, % Cyr
 7705 N = {Ჺ,᲻,᲼,Ჽ,Ჾ,Ჿ,᳀,᳁,
 7706 И,Й,Ѡ,ѡ,Ѣ,ѣ}, % Cyr
 7707 O = {Ò,Ó,Ô,Õ,Ö,Ø,Ǫ,ǫ,Ǭ,ǭ,Ƿ,Ǹ,ǹ,ǽ,ǿ,
 7708 O,Θ,Ϝ,ϝ,Ϟ, % Cyr
 7709 Θ}, % Greek
 7710 P = {Პ,Ჟ,
 7711 P,Რ}, % Cyr
 7712 Q = {Q}, % Cyr
 7713 R = {Ტ,Უ,Ფ,Ქ,Ღ,Ყ,Შ,Ჩ},
 7714 S = {Ჰ,Ჱ,Ჲ,Ჳ,Ჴ,Ჵ,Ჶ,Ჷ,
 7715 S}, % Cyr
 7716 T = {Ჸ,Ჹ,Ჺ,᲻,᲼,Ჽ,
 7717 T,Ჿ}, % Cyr
 7718 U = {Û,ú,û,ü,ũ,ū,ǔ,ǖ,Ǘ,Ǚ,ǚ,Ǜ,
 7719 V = {Ṽ,Ṽ},
 7720 W = {Ŵ,ŵ,Ŷ,ŷ,Ÿ,Ź,
 7721 W}, % Cyr
 7722 X = {X,Ი,
 7723 X,Ლ,Ნ}, % Cyr
 7724 Y = {Ÿ,Ź,Რ,Ს,Ტ,Უ,Ფ,Ქ,
 7725 Y,Შ}, % Cyr
 7726 Z = {Ღ,Ყ,Შ,Ჩ,Ც,Ძ},
 7727 a = {à,á,â,ã,ä,å,ą,ą̄,ą̇,ą̈,ą̉,ą̊,ą̋,
 7728 a,ă,ă̄}, % Cyr
 7729 æ = {æ,
 7730 æ}, % Cyr
 7731 b = {b,Ბ,Გ},
 7732 c = {ç,ć,ĉ,ċ,č,č̣,
 7733 c,ç}, % Cyr
 7734 d = {d,Დ,Ე,Ვ,Ზ},
 7735 e = {è,é,ê,ë,ē,ě,ē,ē,ē,ē,ē,ē,ē,ē,
 7736 e,è,ě,ě}, % Cyr
 7737 f = {f,ff}, % /f_f
 7738 g = {ğ,ğ̇,ğ̈,ğ̉,ğ̊,ğ̋},
 7739 h = {h,Თ,Ი,Კ,Ლ,Მ,Ნ,Ო,
 7740 h,h}, % Cyr
 7741 i = {i,Კ,Ლ,Მ,Ნ,Ო,Პ,Ჟ,Რ,Ს,
 7742 i,ī}, % Cyr
 7743 j = {j,Უ},
 7744 j}, % Cyr
 7745 k = {k,Ვ,Ზ,Თ,Ი},
 7746 l = {l,Თ,Ი,Კ,Ლ}, % l,l
 7747 m = {m,Დ,Ე},

7800 % missing: tipa, math, symbols, ...

```
7801 </CharisSIL>
7802 <*PalatinoLinotype>
7803 \DeclareCharacterInheritance
7804 { encoding = {EU1,EU2},
7805   family = {PalatinoLinotype} }
```

Unfortunately, I don't have a Palatino variant containing all of the following glyphs, which will therefore not show up here. The following is typeset in T_EX Gyre Pagella. To see the real settings, consult mt-PalatinoLinotype.cfg.

```
7806 { A = {À,Á,Â,Ã,Ä,Å,Ā,Ă,Ą,Ǽ,,,,,Ǻ,ǻ,Ǽ,ǽ,ǿ,Ǿ,ǿ,ǿ,ǿ,ǿ,ǿ,ǿ},
7807   B = {,},
7808   C = {Ç,Ć,Ĉ,Č,Ċ},
7809   D = {Ď,,D,,D,,},
7810   E = {È,É,Ê,Ë,Ĕ,Ė,Ě,Ě,Ě,Ě,Ě,Ě,,,,,,E,Ê,Ë,Ê,Ê,Ê,Ê},
7811   F = {},
7812   G = {Ĝ,Ğ,Ġ,Ģ,Ģ,Ģ},
7813   H = {Ĥ,,H,,H,,H},
7814   I = {Ì,Í,Î,Ï,Ì,Í,Î,Ï,Ì,Í,Î,Ï,Ì,Ï},
7815   J = {Ĵ},
7816   K = {K,,,,},
7817   L = {Ĺ,L,L,L,Ļ,,,,L,L}, % L·
7818   M = {,M},
7819   N = {Ñ,Ń,N,Ń,,Ñ,N,,},
7820   O = {Ò,Ó,Ô,Õ,Ö,Ŏ,Ŏ,Ŏ,Ŏ,Ŏ,Ŏ,Ŏ,,,,,,O,Ó,Ó,Ò,Ó,Ŏ,Ŏ,Ŏ,Ŏ,Ŏ,Ŏ},
7821   P = {,},
7822   R = {Ŕ,Ŗ,Ř,Ŗ,,Ŗ,R,Ŕ},
7823   S = {Ś,Ŝ,Ş,Ş,,S,,},
7824   T = {Ţ,Ť,Ŧ,,T,T},
7825   U = {Û,Ú,Ū,Ŭ,Ū,Ŭ,Ū,Ŭ,Ū,Ŭ,Ū,Ŭ,,,,,,U,Ū,Ū,Ū,Ū,Ū,Ū},
7826   V = {,},
7827   W = {Ŵ,Ŷ,Ŵ,Ŵ,,},
7828   X = {,},
7829   Y = {Ý,Ÿ,Ÿ,,ÿ,ÿ,ÿ,ÿ},
7830   Z = {Ž,Ż,Ż,,Z},
7831   a = {à,á,â,ã,ä,å,ā,ă,ą,ǻ,,,,,Ǻ,ǻ,Ǽ,ǽ,ǿ,Ǿ,ǿ,ǿ,ǿ,ǿ,ǿ}, % a²
7832   b = {,},
7833   c = {ç,ć,ĉ,č,ċ},
7834   d = {ď,,d,,},
7835   e = {è,é,ê,ë,ē,ě,é,ę,ě,è,,,,,,e,ê,ë,ê,ê,ê,ê},
7836   f = {ff},
7837   g = {ĝ,ğ,ġ,ğ,ğ,ğ},
7838   h = {ĥ,,h,,h,,},
7839   i = {ì,í,î,ï,ì,í,î,ï,ì,í,î,ï,ì,ï},
7840   j = {ĵ},
7841   k = {k,,,,},
7842   l = {ĺ,,l,,l,,}, % l·
7843   m = {,m},
7844   n = {ñ,ń,n,ñ,,ñ,n,,}, % n
7845   o = {ò,ó,ô,õ,ö,ŏ,ŏ,ŏ,ŏ,ŏ,ŏ,ŏ,,,,,,O,ó,ò,ô,ô,ô,ô,ó,ó,ó,ó},
7846   p = {,},
7847   r = {ŕ,ŗ,ř,,ŕ,ŗ,ŗ},
```


7898 : = { ,500},
 7899 ; = { ,500},
 7900 ! = { ,100},
 7901 ? = { ,200},
 7902 @ = {50,50},
 7903 ~ = {200,250},
 7904 \% = {50,50},
 7905 * = {300,300},
 7906 + = {250,250},
 7907 - = {400,500}, % /hyphen
 7908 – = {400,300}, % /endash
 7909 — = {300,200}, % /emdash
 7910 _ = {200,200}, % /underscore
 7911 / = {200,300},
 7912 /backslash = {200,300},
 7913 ' = {300,400}, % /quotesingle
 7914 ‘ = {500,700}, ’ = {500,600},
 7915 “ = {500,300}, ” = {200,600},
 7916 , = {400,400}, ,, = {400,400},
 7917 ‹ = {400,400}, › = {300,500},
 7918 « = {300,200}, » = {100,400},
 7919 ï = {100, }, ÿ = {100, },
 7920 (= {300, },) = { ,300},
 7921 < = {200,100}, > = {100,200},
 7922 /braceleft = {400,200}, /braceright = {200,400},
 7923 /angleleft = {400, }, /angleright = { ,400},
 7924 † = {100,100},
 7925 ‡ = { 80, 80},
 7926 • = {200,200},
 7927 · = {400,450}, % / periodcentered
 7928 °C = { 80, 50},
 7929 ™ = { , 50},
 7930 ° = {400,400},
 7931 ™ = {100,200},
 7932 © = {100,100},
 7933 ® = {100,100},
 7934 ª = {100,200},
 7935 º = {100,200},
 7936 ¹ = {200,250},
 7937 ² = { 50,100},
 7938 ³ = { 50,100},
 7939 ¬ = {200, },
 7940 − = {300,300},
 7941 ± = {150,200},
 7942 × = {150,250},
 7943 ÷ = {150,250},
 7944 € = {100, },
 7945 /one.oldstyle = {100,100},
 7946 /two.oldstyle = { 50, 50},
 7947 /three.oldstyle = { 30, 80},
 7948 /four.oldstyle = { 50, 50},
 7949 /seven.oldstyle = { 50, 80},

```

7950   Γ = {  ,180}, % /Gamma
7951   Δ = {100,100}, % /Delta
7952   Θ = { 50, 50}, % /Theta
7953   Λ = {100,100}, % /Lambda
7954 %   Ξ = {,}, % /Xi
7955 %   Π = {,}, % /Pi
7956   Σ = { 50, 50}, % /Sigma
7957   Υ = {100,100}, % /Upsilon
7958   Φ = { 50, 50}, % /Phi
7959   Ψ = { 50, 50}, % /Psi
7960 %   Ω = {,}, % /Omega
7961   }
7962
7963 \SetProtrusion
7964   [ name = LMR-it ]
7965   { encoding = {EU1,EU2},
7966     family = Latin Modern Roman,
7967     shape = {it,s1} }
7968   {
7969     A = {125,100},
7970     Æ = {125,-55},
7971     B = {90,-40},
7972     C = {145,-75},
7973     D = {75, -28},
7974     E = {80,-55},
7975     F = {85,-80},
7976     G = {153,-15},
7977     H = {73,-60},
7978     I = {140,-120},
7979     IJ = {140,-80},
7980     J = {135,-80},
7981     K = {70,-30},
7982     L = {87, 40},
7983     M = {67,-45},
7984     N = {75,-55},
7985     O = {150,-30},
7986     Œ = {150,-55},
7987     P = {82,-50},
7988     Q = {150,-30},
7989     R = {75, 15},
7990     S = {90,-65},
7991     $ = {100,-20},
7992     T = {220,-85},
7993     U = {230,-55},
7994     V = {260,-60},
7995     W = {185,-55},
7996     X = {70,-30},
7997     Y = {250,-60},
7998     Z = {90,-60},
7999     a = {150,-10},
8000     b = {170, },
8001     c = {173,-10},
8002     d = {150,-55},

```

8003 e = {180, },
8004 f = { , -250},
8005 g = {150, -10},
8006 h = {100, },
8007 i = {210, },
8008 ij = {210, -40},
8009 j = { , -40},
8010 k = {110, -50},
8011 l = {240, -110},
8012 m = {80, },
8013 n = {115, },
8014 o = {155, },
8015 q = {170, -40},
8016 r = {155, -40},
8017 s = {130, },
8018 t = {230, -10},
8019 u = {120, },
8020 v = {140, -25},
8021 w = {98, -20},
8022 x = {65, -40},
8023 y = {130, -20},
8024 z = {110, -80},
8025 0 = {170, -85},
8026 1 = {230, 110},
8027 2 = {130, -70},
8028 3 = {140, -70},
8029 4 = {130, 80},
8030 5 = {160, },
8031 6 = {175, -30},
8032 7 = {250, -150},
8033 8 = {130, -40},
8034 9 = {155, -80},
8035 . = { , 500},
8036 {,} = { , 450},
8037 : = { , 300},
8038 ; = { , 300},
8039 & = {130, 30},
8040 \% = {180, 50},
8041 * = {380, 20},
8042 + = {180, 200},
8043 @ = {180, 10},
8044 ~ = {200, 150},
8045 (= {300, },) = { , 70},
8046 / = {100, 100},
8047 - = {500, 300}, % /hyphen
8048 – = {500, 300}, % /endash
8049 — = {400, 170}, % /emdash
8050 _ = {100, 200}, % /underscore
8051 ' = {300, 400}, % /quotesingle
8052 " = {500, 300},
8053 ‘ = {800, 200}, ’ = {800, -20},
8054 “ = {540, 100}, ” = {500, 100},

```

8055   , = {300,700}, ,, = {200,600},
8056   ‹ = {500,300}, › = {400,400},
8057   « = {400,100}, » = {200,300},
8058   ¡ = {200, }, ì = {200, },
8059   < = {300,100}, > = {200,100},
8060   /backslash = {300,300},
8061   /braceleft = {400,100}, /braceright = {200,200},
8062   † = {200, 80},
8063   ‡ = {120, 80},
8064   • = {220,100},
8065   · = {550,300}, % / periodcentered
8066   °C = {170, },
8067   ℄ = {100, 50},
8068   ¶ = {200, },
8069   ° = {500,300},
8070   ™ = {200, 70},
8071   © = { 50, 70},
8072   ® = { 50, 70},
8073   ª = {140,100},
8074   º = {140,100},
8075   ¹ = {400,150},
8076   º = {250, 80},
8077   ³ = {250, 80},
8078   ¬ = {250, 80},
8079   − = {300,200},
8080   ± = {150,170},
8081   × = {200,200},
8082   ÷ = {200,200},
8083   € = {150, },
8084   /one.oldstyle = {100,100},
8085   /two.oldstyle = {100, 80},
8086   /three.oldstyle = { 80, 50},
8087   /four.oldstyle = { 80, 80},
8088   /five.oldstyle = { 50, },
8089   /six.oldstyle = { 50, },
8090   /seven.oldstyle = { 80, 80},
8091   /eight.oldstyle = { 50, },
8092   Γ = {100,120}, % /Gamma
8093   Δ = {120,100}, % /Delta
8094   Θ = {120, 50}, % /Theta
8095   Λ = {130,100}, % /Lambda
8096   Ξ = {100,}, % /Xi
8097   Π = {100,}, % /Pi
8098   Σ = {100, 50}, % /Sigma
8099   Υ = {180,100}, % /Upsilon
8100   Φ = {130, 70}, % /Phi
8101   Ψ = {130, 50}, % /Psi
8102   Ω = { 50,}, % /Omega
8103   }
8104   (/LatinModernRoman)
8105   (*CharisSIL)
8106   \SetProtrusion
8107   [ name = Charis-default ]

```

```
8108 { encoding = {EU1,EU2},
8109   family   = Charis SIL }
8110 {
8111   A = {50,50},
8112   Æ = {50,50},
8113   C = {50, },
8114   D = { ,50},
8115   F = { ,50},
8116   G = {50, },
8117   J = {100, },
8118   K = { ,50},
8119   L = { ,50},
8120   Ḷ = { ,100},
8121   O = {50,50},
8122   Œ = {50, },
8123   P = { ,50},
8124   Q = {50,70},
8125   R = { ,50},
8126   ß = { ,40}, % capital sharp s
8127   T = {50,50},
8128   V = {50,50},
8129   W = {50,50},
8130   X = {50,50},
8131   Y = {50,50},
8132   k = { ,50},
8133   ḷ = { ,150},
8134   r = { ,50},
8135   t = { ,50},
8136   v = {50,50},
8137   w = {50,50},
8138   x = {50,50},
8139   y = { ,50},
8140   1 = {150,150},
8141   2 = {50,50},
8142   3 = {50, },
8143   4 = {100,50},
8144   6 = {50, },
8145   7 = {50,80},
8146   9 = {50,50},
8147   . = { ,600},
8148   {,} = { ,500},
8149   : = { ,400},
8150   ; = { ,300},
8151   ! = { ,100},
8152   ? = { ,200},
8153   @ = {50,50},
8154   ~ = {200,250},
8155   \% = { ,50},
8156   * = {300,300},
8157   + = {200,250},
8158   / = { ,200},
8159   /backslash = {150,200},
```


8160 | = {200,200},
8161 - = {400,500}, % hyphen
8162 – = {200,300}, % endash
8163 — = {150,250}, % emdash
8164 — = {200,200}, % Horizontal Bar = \texttwelveudash
8165 - = {150,150}, % Figure Dash = \textthreequartersemdash
8166 _ = {100,100},
8167 {=} = {100,100},
8168 ‘ = {300,400}, ’ = {300,400},
8169 “ = {300,300}, ” = {300,300},
8170 , = {400,400}, „ = {300,300},
8171 ‹ = {400,300}, › = {300,400},
8172 « = {200,200}, » = {150,300},
8173 ï = {100, }, ÿ = {100, },
8174 (= {200, },) = { ,200},
8175 < = {200,150}, > = {100,200},
8176 [= {100, },] = { ,100},
8177 /braceleft = {200, }, /braceright = { ,300},
8178 † = { 80, 80},
8179 ‡ = {100,100},
8180 • = {200,200},
8181 ° = {150,200},
8182 ™ = {150,150},
8183 ¤ = { 50, },
8184 £ = { 50, },
8185 † = {200,200},
8186 © = {100,100},
8187 ® = {100,100},
8188 ª = {100,200},
8189 º = {200,200},
8190 ¬ = {200, 50},
8191 μ = { ,100},
8192 ¶ = { ,100},
8193 · = {300,400},
8194 ¹ = {200,300},
8195 º = {100,200},
8196 ³ = {100,200},
8197 € = {100, },
8198 ± = {150,200},
8199 × = {200,200},
8200 ÷ = {250,250},
8201 /minus = {200,200},
8202 − = {200,200},
8203 % Cyrillic
8204 Б = { ,50},
8205 Г = { ,130},
8206 Ж = {50,50},
8207 З = {30,50},
8208 Л = {50, },
8209 У = {50,50},
8210 Ф = {50,50},
8211 Ч = {100, },

8212 Ъ = { ,50},
8213 б = { ,50},
8214 Э = {50,50},
8215 Ю = { ,40},
8216 Я = {50, },
8217 V = {50,50},
8218 Ė = {50, },
8219 Ъ = {50,100},
8220 Ć = {50, },
8221 Љ = {50,50},
8222 Њ = { ,50},
8223 Ћ = {50,50},
8224 Ѻ = {100,100},
8225 ǰ = {50,50},
8226 Ъ = { ,50},
8227 б = { ,50},
8228 Љ = {50,80},
8229 Њ = { ,80},
8230 Ћ = {50,50},
8231 ЈЈ = {50, },
8232 ЈХ = {50,40},
8233 К = { ,50},
8234 Æ = {50, },
8235 Љ = { ,50},
8236 Њ = { ,50},
8237 đ = { ,100},
8238 б = {50,50},
8239 г = { ,70},
8240 к = { ,50},
8241 л = {50, },
8242 т = {50,50},
8243 ф = {50,50},
8244 ч = {50, },
8245 ъ = { ,50},
8246 ь = { ,50},
8247 э = { ,50},
8248 я = {50, },
8249 љ = {50, },
8250 њ = { ,50},
8251 ы = { ,50},
8252 v = {50,50},
8253 ė = {50, },
8254 ʃ = { ,50},
8255 ʄ = {50,50},
8256 ʅ = { ,50},
8257 ʆ = { ,50},
8258 đ = { ,100},
8259 Ѻ = {100,100},
8260 ǰ = {50,50},
8261 ѻ = {50,70},
8262 Ѽ = { ,70},
8263 æ = {50,30},

```

8264   Ъ = { ,50},
8265   Ы = { ,50},
8266   %   Д П Ц Ш Щ Ъ Ы Ь Ѡ ѡ Ѣ ѣ Ѡ ѡ ѣ ѣ
8267   %   в д ж з и м н п ц ш ы ю ѣ ѣ ѣ ѣ ѣ ѣ ѣ ѣ ѣ ѣ
8268   % Greek
8269   Δ = {50,50},
8270   Ψ = {50,50},
8271   γ = {70,70},
8272   λ = {40,70},
8273   π = {40,50},
8274   ρ = { ,50},
8275   σ = { ,50},
8276   χ = {50,50},
8277 }
8278
8279 \SetProtrusion
8280 [ name      = Charis-it   ]
8281 { encoding = {EU1,EU2},
8282   family   = Charis SIL,
8283   shape    = {it,sl} }
8284 {
8285   C = {50, },
8286   G = {50, },
8287   J = {50, },
8288   L = {50,50},
8289   O = {50, },
8290   Œ = {50, },
8291   Q = {50, },
8292   S = {50, },
8293   $ = {50, },
8294   T = {70, },
8295   o = {50,50},
8296   p = { ,50},
8297   q = {50, },
8298   t = { ,50},
8299   w = { ,50},
8300   y = { ,50},
8301   1 = {150,100},
8302   3 = {50, },
8303   4 = {100, },
8304   6 = {50, },
8305   7 = {100, },
8306   . = { ,700},
8307   {,} = { ,600},
8308   : = { ,400},
8309   ; = { ,400},
8310   ? = { ,150},
8311   & = { ,80},
8312   \% = {50,50},
8313   * = {300,200},
8314   + = {250,250},
8315   @ = {80,50},
8316   ~ = {150,150},

```

8317 / = { ,150},
8318 /backslash = {150,150},
8319 - = {300,400}, % hyphen
8320 – = {200,300}, % endash
8321 — = {150,200}, % emdash
8322 _ = { ,100},
8323 {=} = {200,200},
8324 ± = {150,200},
8325 × = {250,250},
8326 ÷ = {250,250},
8327 ° = {150,200},
8328 · = {300,400},
8329 ‘ = {400,200}, ’ = {400,200},
8330 “ = {300,200}, ” = {400,200},
8331 , = {200,500}, ,, = {150,500},
8332 ‹ = {300,400}, › = {200,500},
8333 « = {200,300}, » = {150,400},
8334 (= {200, },) = { ,200},
8335 < = {200,200}, > = {200,200},
8336 /braceleft = {300, }, /braceright = { ,200},
8337 % Cyrillic
8338 Ж = {50,30},
8339 Л = {50, },
8340 У = {50,30},
8341 Ф = {50, },
8342 Ч = {100, },
8343 Ъ = { ,50},
8344 Ь = { ,50},
8345 Э = {50,50},
8346 Я = {50, },
8347 V = {50,50},
8348 Љ = {50,50},
8349 Ђ = {140,100},
8350 Ѓ = {70,50},
8351 Љ = {50,80},
8352 Њ = { ,80},
8353 Ћ = {50,50},
8354 г = {50,50},
8355 д = {50,30},
8356 м = {50, },
8357 ф = {50, },
8358 ч = {50, },
8359 ъ = { ,50},
8360 ь = { ,50},
8361 э = { ,50},
8362 я = {50, },
8363 љ = {50,50},
8364 њ = { ,50},
8365 в = {50,50},
8366 ы = { ,50},
8367 ф = {140,100},
8368 х = {70,50},

```

8369   Ϛ = {50,70},
8370   ϛ = { ,70},
8371 % Greek
8372   Γ = { ,130},
8373   Δ = {50,50},
8374   Ψ = {50,50},
8375   γ = {70,70},
8376   λ = {40,70},
8377   π = {40,50},
8378   ρ = { ,50},
8379   σ = { ,50},
8380   χ = {50,50},
8381   }
8382
8383 \SetProtrusion
8384   [ name      = Charis-sc,
8385     load      = Charis-default ]
8386   { encoding = {EU1,EU2},
8387     family   = Charis SIL,
8388     shape    = {sc} }
8389   {
8390   % A = {100,100}, % etc., doesn't work with \textsc
8391   /a.SC = {100,100},
8392   /c.SC = {50, },
8393   /d.SC = { ,50},
8394   /f.SC = { ,50},
8395   /g.SC = {50, },
8396   /j.SC = {100, },
8397   /k.SC = { ,50},
8398   /l.SC = { ,50},
8399   /f_l.SC = { ,50},
8400   /o.SC = {50,50},
8401   /oe.SC = {50, },
8402   /q.SC = {50,70},
8403   /r.SC = { ,50},
8404   /t.SC = {50,100},
8405   /v.SC = {50,50},
8406   /w.SC = {50,50},
8407   /x.SC = {50,50},
8408   /y.SC = {50,50}
8409   }
8410 </CharisSIL>
8411 <*PalatinoLinotype>
8412 \SetProtrusion
8413   [ name      = palatino-default ]
8414   { encoding = {EU1,EU2},
8415     family   = {PalatinoLinotype} }
8416   {
8417   A = {50,50},
8418   D = { ,50},
8419   J = {50, },
8420   K = { ,50},
8421   L = { ,50},
8422   O = {25, },

```

```

8423 T = {50,50},
8424 V = {50,50},
8425 W = {50,50},
8426 X = {50,50},
8427 Y = {50,50},
8428 b = { ,25},
8429 d = {25,30},
8430 f = { ,50},
8431 g = { ,100},
8432 k = { ,50},
8433 p = { ,50},
8434 q = {50, },
8435 r = { ,50},
8436 t = { ,50}, = { ,50}, = { ,50},
8437 v = {75,50},
8438 w = {50,50},
8439 x = {50,50},
8440 y = {50,70},
8441 1 = {100,50},
8442 2 = {25,50},
8443 4 = {50, },
8444 6 = {50, },
8445 9 = {25, },
8446 Æ = {100, },
8447 Œ = {25, },
8448 . = { ,700}, = { ,350}, ... = { ,150},
8449 {,} = { ,500},
8450 := { ,500},
8451 ; = { ,500},
8452 != { ,100}, = { ,100},
8453 ? = { ,200}, ? = { ,200},
8454 @ = {50,50},
8455 ~ = {200,250},
8456 & = {50,100},
8457 \% = {100,100},
8458 * = {200,200},
8459 + = {250,250},
8460 (= {100, }, ) = { ,300},
8461 / = {200,300},
8462 - = {400,500},
8463 \textendash = {300,300}, \textemdash = {200,200},
8464 \textquoteleft = {500,700}, \textquoteright = {500,700},
8465 \textquotedblleft = {300,400}, \textquotedblright = {300,400},
8466 \textbackslash = {200,300},
8467 \quotesinglbase = {400,400}, \quotedblbase = {400,400},
8468 \guilsinglleft = {400,400}, \guilsinglright = {300,500},
8469 \guillemotleft = {300,300}, \guillemotright = {200,400},
8470 \textexclamdown = {100, }, \textquestiondown = {100, },
8471 \textbraceleft = {400,200}, \textbraceright = {200,400},
8472 \textless = {200,100}, \textgreater = {100,200},
8473 ≤ = {200,100}, ≥ = {100,200},
8474 \textminus = {300,300},

```

```

8475 \texttrademark      = {200,200},
8476 \textcopyright      = {200,200},
8477 \textregistered     = {200,200},
8478 \textdegree         = {300,300},
8479 ¡                  = {450,500}, ¯          = {250,150},
8480                  = {150,250},
8481 ·                  = {850, 700},
8482 ¶                  = {100,0},
8483 ×                  = {150, 300},
8484 ª                  = {300,300}, º          = {300,300},
8485                  = {200,400},
8486 ¹ = {400,350},    º = {200,300},     = {250,400},
8487                  = {250,350},    = {200,300},    = {250,400},
8488                  = {200,450},    = {250,400},    = {200,350},
8489                  = {200,400},
8490                  = {400,250},    = {200,300},    = {250,400},
8491                  = {250,350},    = {200,300},    = {250,400},
8492                  = {200,450},    = {250,400},    = {200,350},
8493 ± = {150,100},    ÷ = {300,300},
8494 þ = { ,25},
8495                  = {300,450},    = {300,450},
8496                  = {300,450},    = {300,450},
8497 †                  = {200,250}, ‡          = {200,250},
8498 π = {50, },
8499 f = { ,50},
8500 № = {100,150},
8501 \textservicemark    = {100,200},
8502 - = {400,500},    - = {400,500},    = {200,300},
8503 - = {205,305},    — = {200,300},    = {50,150},
8504 • = {125,200},
8505 % /a.sc = {50,50},
8506 }
8507
8508 \SetProtrusion
8509 [ name = palatino-it ]
8510 { encoding = {EU1,EU2},
8511   family = {PalatinoLinotype},
8512   shape = {it,sl} }
8513 {
8514 A = {50,50},
8515 Æ = {50, },
8516 B = {50, },
8517 C = {50, },
8518 D = {50,50},
8519 E = {50, },
8520 F = {50, },
8521 G = {50, },
8522 H = {50, },
8523 K = {50, },
8524 L = {50, },
8525 O = {50, },
8526 Œ = {50, },
8527 P = {50, },

```

```

8528 Q = {50, },
8529 R = {50, },
8530 S = {50, },
8531 $ = {50, },
8532 T = {100, },
8533 U = {50, },
8534 V = {100,50},
8535 W = {50, },
8536 X = {50, },
8537 Y = {100,50},
8538 b = { ,50},
8539 c = {25, },
8540 g = {75, },
8541 i = {25, },
8542 m = { ,50},
8543 n = { ,50},
8544 p = { ,25},
8545 q = {25, },
8546 x = { ,50},
8547 1 = {100, },
8548 2 = {50, },
8549 4 = {50, },
8550 7 = {50, },
8551 . = { ,500},    = { ,350},    ... = { ,200},
8552 {,}= { ,500},
8553 := { ,300},
8554 ; = { ,300},
8555 ? = { ,300},    ? = { ,300},
8556 & = {50,50},
8557 \% = {100,100},
8558 * = {200,200},
8559 + = {150,200},
8560 @ = {50,50},
8561 ~ = {200,150},
8562 (= {200, }, ) = { ,200},
8563 / = {100,200},
8564 - = {300,500},
8565 \textendash    = {300,300}, \textemdash    = {200,200},
8566 \textquoteleft = {700,400}, \textquoteright = {700,400},
8567 \textquotedblleft = {500,300}, \textquotedblright = {500,300},
8568 _ = {100,100},
8569 \textbackslash = {100,200},
8570 \quotesinglbase = {500,500}, \quotedblbase = {400,400},
8571 \guilsinglleft = {400,400}, \guilsinglright = {300,500},
8572 \guillemotleft = {300,300}, \guillemotright = {300,300},
8573 \textexclamdown = {100, }, \textquestiondown = {200, },
8574 \textbraceleft = {200,100}, \textbraceright = {200,200},
8575 \textless      = {300,100}, \textgreater      = {200,100},
8576 ≤            = {200,100}, ≥            = {100,200},
8577 †            = {450,500}, ¬            = {250,150},
8578 ·            = {850,700},
8579 ¶            = {100,0},

```



```

8580   ×                = {150, 300},
8581   ª = {300,250},    ° = {300,300},    º = {300,250},
8582   = {300,200},
8583   ¹ = {300,150},    ² = {350,200},    ³ = {250,150},
8584   = {350,100},    = {300, 50},    = {400,100},
8585   = {400, 50},    = {250, 50},    = {300, 50},
8586   = {300,300},
8587   = {300,350},    = {300,150},    = {250,250},
8588   = {400,200},    = {300,100},    = {450,200},
8589   = {450,150},    = {400,250},    = {400,200},
8590   ± = {150,100},    ÷ = {300,300},
8591   þ = { 50, },
8592   †                = {250,200}, ‡                = {250,200},
8593   = {300,450},    = {300,450},
8594   = {300,450},    = {300,450},
8595   - = {300,500},    - = {300,500},    = {100,300},
8596   - = {125,305},    — = {200,300},    = {125,150},
8597   • = {125,200}
8598   }
8599
8600 \SetProtrusion
8601   [ name      = palatino-sc,
8602     load      = palatino-default ]
8603   { encoding = {EU1,EU2},
8604     family   = {PalatinoLinotype},
8605     shape    = sc }
8606   {
8607     a = {50,50},
8608     æ = {50, },
8609     b = { 0, 0},
8610     d = { 0, 0},
8611     f = { 0, 0},
8612     g = { 0, 0},
8613     j = {50, },
8614     l = { ,50},
8615     o = { 0, 0},
8616     p = { 0, 0},
8617     q = { 0, },
8618     r = { , 0},
8619     t = {50,50},
8620     y = {50,50},
8621     fl = { 0,50},
8622     ffl = { 0,50},
8623     = { 0,50},
8624     = { 0,50}
8625   }
8626 </PalatinoLinotype>
8627

```

17 Auxiliary file for micro fine tuning

This file can be used to test protrusion and expansion settings.

```

8628 (*test)
8629 \documentclass{article}
8630
8631 %% Here you can specify the font you want to test, using
8632 %% the commands \fontfamily, \fontseries and \fontshape.
8633 %% Make sure to end all lines with a comment character!
8634 \newcommand*\TestFont{%
8635   \fontfamily{ppl}%
8636   %% \fontseries{b}%
8637   %% \fontshape{it}% sc, sl
8638 }
8639
8640 \usepackage{ifthen}
8641 \usepackage[T1]{fontenc}
8642 \usepackage[latin1]{inputenc}
8643 \usepackage[verbose,expansion=alltext,stretch=50]{microtype}
8644
8645 \pagestyle{empty}
8646 \setlength{\parindent}{0pt}
8647 \newcommand*\crulefill{\cleaders\hbox{$\mkern-2mu\smash-\mkern-2mu$}\hfill}
8648 \newcommand*\testprotrusion[2][ ]{%
8649   \ifthenelse{\equal{#1}{r}}{\#2}%
8650   lorem ipsum dolor sit amet,
8651   \ifthenelse{\equal{#1}{r}}{\crulefill}{\leftarrowfill} #2
8652   \ifthenelse{\equal{#1}{l}}{\crulefill}{\rightarrowfill}
8653   you know the rest%
8654   \ifthenelse{\equal{#1}{l}}{\#2}%
8655   \linebreak
8656   {\fontencoding{\encodingdefault}%
8657   \fontseries{\seriesdefault}%
8658   \fontshape{\shapedefault}%
8659   \selectfont
8660   Here is the beginning of a line, \dotfill and here is its end}\linebreak
8661 }
8662 \newcommand*\showTestFont{\expandafter\stripprefix\meaning\TestFont}
8663 \def\stripprefix#1>{}
8664 \newcount\charcount
8665 \begin{document}
8666
8667 \microtypesetup{expansion=false}
8668
8669 {\centering The font in this document is called by:\\
8670 \texttt{\showTestFont}\par}\bigskip
8671
8672 \TestFont\selectfont
8673 This line intentionally left empty\linebreak
8674 %% A -- Z
8675 \charcount=65
8676 \loop
8677   \testprotrusion{\char\charcount}
8678   \advance\charcount 1
8679   \ifnum\charcount < 91 \repeat
8680 %% a -- z
8681 \charcount=97
8682 \loop
8683   \testprotrusion{\char\charcount}
8684   \advance\charcount 1
8685   \ifnum\charcount < 123 \repeat
8686 %% 0 -- 9
8687 \charcount=48
8688 \loop

```

```
8689 \testprotrusion{\char\charcount}
8690 \advance\charcount 1
8691 \ifnum\charcount < 58 \repeat
8692 %%
8693 \testprotrusion[r]{,}
8694 \testprotrusion[r]{.}
8695 \testprotrusion[r]{;}
8696 \testprotrusion[r]{:}
8697 \testprotrusion[r]{?}
8698 \testprotrusion[r]{!}
8699 \testprotrusion[l]{\textexclamdown}
8700 \testprotrusion[l]{\textquestiondown}
8701 \testprotrusion[r]{\{ }
8702 \testprotrusion[l]{\{ (}
8703 \testprotrusion{/}
8704 \testprotrusion{\char~\}
8705 \testprotrusion{-}
8706 \testprotrusion{\textendash}
8707 \testprotrusion{\textemdash}
8708 \testprotrusion{\textquoteleft}
8709 \testprotrusion{\textquoteright}
8710 \testprotrusion{\textquotedblleft}
8711 \testprotrusion{\textquotedblright}
8712 \testprotrusion{\quotesinglbase}
8713 \testprotrusion{\quotedblbase}
8714 \testprotrusion{\guilsinglleft}
8715 \testprotrusion{\guilsinglright}
8716 \testprotrusion{\guillemotleft}
8717 \testprotrusion{\guillemotright}
8718
8719 \newpage
8720 The following displays the current font stretched by 5%,
8721 normal, and shrunk by 5%:
8722
8723 \bigskip
8724 \newlength{\MTln}
8725 \newcommand*{\teststring}
8726 {ABCDEFGHIJKLMNQRSTUvwxyz0123456789}
8727 \settowidth{\MTln}{\teststring}
8728 \microtypesetup{expansion=true}
8729
8730 \parbox{1.05\MTln}{\teststring\linebreak\}
8731 \parbox{0.95\MTln}{\teststring}\par\bigskip
8732 \parbox{0.95\MTln}{\teststring}
8733
8734 \end{document}
8735 /test
```

Needless to say that things may always be improved. For suggestions, mail to w.m.l@gmx.net.

A The title logo

8736 *(*logo)*

Here's how the logo on the title page was created.³⁰ 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.³¹ 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`.

8737 `\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.

8738 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

8739 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

8740 `\newdimen\fboxrulei`

8741 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

8742 `\newdimen\fboxruleii`

8743 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

8744 `\newdimen\kernboxheight`

8745 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

8746 `\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).

8747 `\fontinstcc`

8748 `\def\showlogo#1{%`

Some fonts do not specify the `\fontdimen6` (width of an `em`) in the `afm` file. In this case, use the font size, which is correct in most cases.

8749 `\ifdim\fontdimen6\font = 0pt`

8750 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%}`

8751 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTeXversion < 130 pt\fi-***}`

8752 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTeXversion < 130 pt\fi\relax`

8753 `\fi`

8754 `\installfonts`

8755 `\input_metrics{}{\logofont,\metrics\printbbs{#1}\relax}`

8756 `\endinstallfonts`

8757 `}`

8758 `\normalcc`

Layers.

8759 `\makeatletter`

8760 `\def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}`

8761 `\ifx\mt@objects\undefined\let\mt@objects\@empty\fi`

8762 `\ifx\mt@order \undefined\let\mt@order \@empty\fi`

30 Note that the logo module will not be created when installing `microtype`. Instead, I've included the source in the PDF file as an attachment. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

31 Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

8763 \xdef\mt@order{\mt@order[(Logo)]}
8764 \let\mtl@resources\@empty
8765 \def\mtl@register#1{%
8766   \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
8767   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdflastobj\space 0 R }
8768   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
8769   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
8770   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
8771 \mtl@register{canvas}
8772 \mtl@register{characters}
8773 \mtl@register{bounding-boxes}
8774 \mtl@register{TeX-boxes}
8775 \xdef\mt@order{\mt@order]}
8776 \global\let\mtl@objects\mt@objects
8777 \ifx\pdfcolorstack\undefined
8778   \pdfcatalog{/OCProperties <<
8779     /OCGs [\mt@objects]
8780     /D << /Order [\mt@order] >> >>}
8781 \fi
8782 \def\togglelayer#1#2{%
8783   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
8784   user{/Subtype/Link
8785     /BS << /Type/Border/W 0 >> /H/0
8786     /A << /S/SetOCGState
8787     /State[/Toggle \csname mtl@#1\endcsname] >>
8788   }#2\pdfendlink
8789 }

```

\printbbs Preparation.

```

8790 \setcommand\printbbs#1{%
8791   \setbox0\hbox{#1}%
8792   \leavevmode
8793   \kern-\fboxrulei

```

The canvas in the natural width of the text minus protrusion, in color bgcolor.

```

8794 \mtl@layer{canvas}{%
8795   \getboundarychars#1\relax
8796   \tempdim=\dimexpr\wd0 - (\scaletoe{\lcode\font\firstchar}+
8797     \scaletoe{\rcode\font\lastchar})\relax
8798   \kern\dimexpr\scaletoe{\lcode\font\firstchar}\relax
8799   \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
8800     \hrule width \tempdim
8801     height \dimexpr\dp0+\ht0+0.15em\relax}%
8802   \kern-\tempdim

```

The baseline, in color blcolor.

```

8803   \vbox{\color{blcolor}%
8804     \hrule width \tempdim
8805     height \fboxrulei}%
8806   }%
8807   \kern-\dimexpr\wd0 -\scaletoe{\rcode\font\lastchar}\relax

```

The string.

```

8808   \printbbs #1\relax\relax
8809 }

```

\getboundarychars Get first . . .

```

8810 \def\getboundarychars#1#2\relax{%
8811   \def\firstchar{~#1}%
8812   \getlastchar#1#2\relax
8813 }

```

\getlastchar . . . and last character.

```

8814 \def\getlastchar#1#2{%
8815   \ifx\relax#2\relax
8816     \def\lastchar{~#1}%
8817   \else
8818     \expandafter\getlastchar

```

```

8819   \fi
8820   #2%
8821 }

\printbss   Loop over all characters of the string.
8822 \def\printbss#1#2#3\relax{%
8823   \ifx\relax#1\relax
8824   \else
8825     \ifx\relax#2\relax
8826       \printbb{#1}{}%
8827     \else
8828       \printbb{#1}{#2}%
8829     \fi
8830   \expandafter\printbss
8831   \fi
8832   #2#3\relax
8833 }

\printbb   Record the kern between the current and the following character, then print the character. \kerning is a fontinst
           command.
8834 \setcommand\printbb#1#2{%
8835   \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
8836   \showboxes{#1}%
           This could be another application.
8837 %   \quad
8838 %   w: \the\scaletoe{\width{#1}},
8839 %   bb: \the\scaletoe{\bbleft{#1}}/%
8840 %   \the\scaletoe{\bbright{#1}},
8841 %   \the\scaletoe{\number\numexpr\width{#1}-\bbright{#1}\relax}
8842 %   h: \height{#1}/\bbtop{#1}, \bbbottom{#1}/\depth{#1}\par
8843 }

\showboxes Print the boxes for character <#1>. This will not work if <#1> is not also the PostScript name of the glyph (e.g., 'comma'
           ≠ ',').
8844 \setcommand\showboxes#1{%
8845   \leavevmode
8846   \color{texcolor}%
           We have to record the width of the glyph.
8847   \setbox0\hbox{\color{textcolor}{#1}}%
8848   \global\tempdim=\wd0\relax
8849   \kern-\fboxrulei
           1. The  $\TeX$  box: Print a frame in color texcolor. This frame shows the glyph as  $\TeX$  sees it.
8850   \mtl@layer{TeX-boxes}{%
8851     \hbox{%
8852       \lower\dimexpr \dp0 + \fboxrulei\relax
8853       \hbox{%
8854         \vbox{%
8855           \hrule height\fboxrulei
8856           \hbox{%
8857             \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
8858             \phantom{\unhcopy0}%
8859             \vrule width\fboxrulei
8860           }%
8861           \hrule height\fboxrulei}}}%
8862     }%
           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.
8863   \kern-\wd0
8864   \mtl@layer{characters}{%
8865     \hbox{\box0}%
8866   }%
           Step back by the amount that the character's bounding box differs from the  $\TeX$  box on the left side.
8867   \kern\dimexpr\scaletoe{\bbleft{#1}}-\tempdim-\fboxrulei\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

8868 \mtl@layer{bounding-boxes}{%
8869   {\color{bbcolor}%
8870   \hbox{%
8871     \lower\dimexpr-\scaletoe{\bbottom{#1}}+\fboxrulei\relax
8872     \hbox{%
8873       \vbox{%
8874         \hrule height\fboxrulei
8875         \hbox to \dimexpr\scaletoe{\numexpr
8876           \bbright{#1}-\bbleft{#1}\relax}+2\fboxrulei\relax{%
8877           \vrule height \dimexpr\scaletoe{\numexpr
8878             \bbtop{#1}-\bbottom{#1}\relax}%
8879             width\fboxrulei
8880             \hfill
8881             \vrule width\fboxrulei}%
8882           \hrule height\fboxrulei}}}%
8883     }%
8884     \kern-\dimexpr\fboxrulei+\fboxrulei\relax
8885   }%

```

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.

```

8886 \kern\scaletoe{\numexpr\width{#1}-\bbright{#1}\relax}%
8887 \mtl@layer{TeX-boxes}{%
8888   {\ifnum\thekern<0
8889     \color{kerncolor}%
8890     \kern\scaletoe{\thekern}%
8891     \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletoe{\thekern}\relax
8892       height \kernboxheight}%
8893     \kern\scaletoe{\thekern}%
8894   }else
8895     \color{texcolor}%
8896     \ifnum\thekern=0 \else
8897       \lower\kernboxheight
8898       \hbox{%
8899         \vbox{%
8900           \hrule height\fboxrulei
8901           \hbox{%
8902             \vrule height \kernboxheight width\fboxrulei
8903             \kern\dimexpr\scaletoe{\thekern}-2\fboxrulei\relax
8904             \vrule width\fboxrulei
8905           }%
8906           \hrule height\fboxrulei}}}%
8907         \fi
8908       \fi
8909     }%
8910   }%
8911   % \kern-\fboxrulei
8912 }

```

```

8913 \newbox\logobox
8914 \def\printlogo{%
8915   \setbox\logobox=\hbox{\vbox{%
8916     \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

8917 \def\logofont{pkpri9e10}
8918 \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
8919 \font\thelogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

8920 %\def\logofont{pplri}
8921 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
8922 %\edef\logofont{\logofont8r}
8923 %\font\thelogofont=\logofont\space at 78pt

```

Load the font.

```

8924 \thelogo font
      Protrusion values (overdone for didactic reasons).
8925 \lcode\font`M=96
8926 \rcode\font`e=46
      Now we can generate the logo.
8927 \pdfliteral direct{/SXS gs}%
8928 \showlogo{Microtype}%
8929 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
8930 % \kern5pt\[\[3\baselineskip]
8931 % \long\def\@makefnmark##1{%
8932 % \leftskip 0pt
8933 % \parindent 0pt
8934 % \everypar{\parindent 0pt}%
8935 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
8936 % \footnotetext[1]{This graphic display on a
8937 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
8938 % their \togglelayer{bounding-boxes}{bounding boxes}
8939 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.}
8940 }%
8941 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
8942 \immediate\pdfobj{<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>}%
8943 \immediate\pdfxform
8944     attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
8945     resources {/Properties <<\mtl@resources>>
8946               /ExtGState << /SXS \the\pdflastobj\space 0 R >>
8947             }
8948     \logobox
8949 % \vskip-2.5\baselineskip
8950 % \leavevmode
8951 % \togglelayer{characters}{%
8952 % \pdfrefxform\pdflastxform
8953 % }%
8954 \pdfannot\logodimens{%
8955   /Subtype/Widget /FT/Btn /T(Logo)
8956   %/F 4 % why did I say this?
8957   /AP << /N \the\pdflastxform\space 0 R >>
8958   /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
8959       /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
8960       /D << /S/SetOCGState /State[/Toggle \cname mtl@bounding-boxes\endcname] >>
8961       /U << /S/SetOCGState /State[/Toggle \cname mtl@TeX-boxes\endcname] >>
8962   >>
8963 }%
8964 \vspace{3\baselineskip}
8965 }
      Our font.
8966 \pdfmapline{+pkpmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmri8a10.pfb}
      Define colours (thered and thegreen are copied from microtype.dtx).
8967 \def\mtdefinicolors{
8968 \definecolor{thered}{rgb}{0.65,0.04,0.07}
8969 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
8970 \colorlet{texcolor}{thegreen!50} % TeX boxes
8971 \colorlet{kerncolor}{texcolor} % negative kerns
8972 \colorlet{bbcolor}{thered!50} % bounding box
8973 \colorlet{bgcolor}{black!8} % canvas
8974 \colorlet{blcolor}{black!50} % baseline
8975 \colorlet{textcolor}{black!40} % text
8976 }
      Use with microtype.dtx
8977 \ifx\documentclass@twoclasseserror
8978 \usepackage[xcdraw]{xcolor}
8979 \mtdefinicolors
8980 \else

```


A.2 Document

Now we can start the document.

```

8981 \documentclass[10pt,a4paper]{ltxdoc}
8982 \providecommand\MakePercentComment{\relax}
8983 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

    Re-use the preamble from microtype.dtx.
8984 \usepackage{microtype-doc}
8985 \usepackage{attachfile}
8986 \makeatletter
8987 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
8988 \makeatother
8989 \begin{document}

    You are currently reading this.
8990 \DocInput{microtype-logo.dtx}

    And here's the logo.
8991 \vfill
8992 \begin{center}
8993   \printlogo \null
8994 \end{center}
8995 \vfill
8996 \expandafter\enddocument
8997 \fi

    That's it.
8998 /logo

```

B The letterspacing illustration

```

8999 \ifx\lssample\undefined
9000 *lssample

```

Upon popular request, here's how I've created the letterspacing illustration.³²

B.1 Macros

Rule width and image height and depth.

```

9001 \makeatletter
9002 \newdimen\lsamount
9003 \newdimen\lsrule
9004 \lsrule=0.2pt
9005 \def\lsheight{8pt}
9006 \def\lsdepth{12pt}

    Our font (Adobe Caslon).
9007 \def\lsfont{\fontfamily{paca}\selectfont}

    Loop over all letters in <#2>, letterspacing them by <#1>.
9008 \def\dols#1#2{\lsamount=#1\relax \dols#2\enddols}
9009 \def\dolss#1#2\enddols{%
9010   \ifx\empty#2\empty\divide\lsamount 2\fi
9011   \ls{#1}%
9012   \ifx\empty#2\empty\else \dols#2\enddols \fi
9013 }

    One tikz picture for each letter.
9014 \def\ls#1{%
9015   \begin{tikzpicture}[remember picture,line width=\lsrule]
9016     \tikzstyle{every node}=[inner sep=0pt]

```

³² Note that the `lssample` module will not be created when installing `microtype`. Instead, I've included the source in the PDF file as an attachment. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you can use the `pdftk` tool.

The bounding box.

```
9017 \mts@layer{stuff}{%
9018 \node[draw=thegrey,
9019 fill=theshade,
9020 outer sep=\lsrule,
9021 anchor=base,
9022 font=\lsfont]{\phantom{#1}};
9023 }
```

The letter.

```
9024 \node[anchor=base,font=\lsfont](#1){#1};
```

Two auxiliary coordinates.

```
9025 \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
9026 \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
9027 \mts@layer{stuff}{%
```

Now draw the normal character width,

```
9028 \draw[color=thered!75,
9029 fill=thered!30,
9030 outer sep=\lsrule]
9031 (#1L) rectangle (#1R);
9032 \ifdim\lsamount>0pt
9033 \path (#1.base east) ++(+.5\lsamount,-6pt) coordinate (#1_ls);
9034 \path (#1R) ++(\lsamount+\lsrule,\lsdepth) coordinate (#1E);
```

and the letter space.

```
9035 \draw[color=thered,
9036 fill=thered!50,
9037 outer sep=\lsrule]
9038 (#1R) ++(\lsrule,+0pt) rectangle (#1E);
9039 \fi
9040 }
9041 \end{tikzpicture}%
9042 \ignorespaces
9043 }
```

Draw the interword space.

```
9044 \def\lssp#1#2#3#4{%
9045 \mts@layer{stuff}{%
9046 \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
9047 \tikzstyle{every draw}=[anchor=bottom]
9048 \coordinate(#1space) at (#2/2,\lsdepth/2);
9049 \coordinate(#1stretch) at (#2+#3/2,+0pt);
9050 \coordinate(#1shrink) at (#2-#4/2,+0pt);
9051 \draw[color=thegreen,fill=thegreen!50,use as bounding box]
9052 (0,0) rectangle ++(+#2,\lsdepth);
9053 \draw[color=thegreen,fill=thegreen!30]
9054 (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
9055 \draw[color=thegreen,fill=thegreen!50]
9056 (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
9057 \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]
9058 (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
9059 \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
9060 (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
9061 \end{tikzpicture}%
9062 }\ignorespaces
9063 }
```

Layers.

```
9064 \def\mts@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
9065 \def\mts@layerx#1#2{\pdfliteral{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral{EMC EMC}}
9066 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
9067 \ifx\mt@order \undefined\let\mt@order \@empty\fi
9068 \xdef\mt@order{\mt@order[(Sheep)]}
9069 \let\mts@resources\@empty
9070 \def\mts@register#1{%
```

```

9071 \immediate\pdfobjj{<< /Type/OCG /Name(#1) >>}
9072 \expandafter\xdef\csname mts@#1\endcsname{\the\pdf\astobj\space 0 R }
9073 \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
9074 \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
9075 \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
9076 \mts@register{stuff}
9077 \mts@register{tracking}
9078 \mts@register{ispace}
9079 \mts@register{ospace}
9080 \mts@register{istretch}
9081 \mts@register{ishrink}
9082 \mts@register{ostretch}
9083 \mts@register{oshrink}
9084 \mts@register{okern}
9085 \mts@register{ligature}
9086 \mts@register{_compatibility}
9087 \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
9088 \newcommand\anchorarrow[1]{%
9089 \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
9090 \newcommand\add@arrow[5][left]{%
9091 \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
9092 \mtsx@layer{#3}{%
9093 \draw[->,thick,color=the#2](#4) to[bend #1] (#5);}%
9094 }

Toggle layer.
9095 \def\toggle@layer#1#2#3{%
9096 \pdfstartlink
9097 user{/Subtype/Link
9098 /BS << /Type/Border/W 0 >> /H/0
9099 % /BS << /Type/Border/W 1 /S/D /D[4 1] >>
9100 % /C[0.7 0.7 0.7] /H/0
9101 /Contents(Click to Toggle!)
9102 /A << /S/SetOCGState
9103 /State[/Toggle \csname mts@#1\endcsname] >>
9104 }%
9105 \rlap{#2}%
9106 {\fboxsep=0pt \fboxrule=0pt
9107 \mtsx@layer{stuff}{%
9108 \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
9109 \mtsx@layer{#1}{%
9110 \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}}%
9111 }%
9112 \pdfendlink
9113 }
9114 \newcommand\showarrow[2][ ]{%
9115 \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
9116 \toggle@layer{\@tempa}{\itshape #2}}

The environment for our illustration.
9117 \def\ls@sample#1{%
9118 \parskip 4pt \parindent 0pt
9119 \par
9120 \vskip4pt
9121 {\leftskip 15pt
9122 \mt@pseudo@marg{\color{theblue}Click on the image to show the kerns
9123 and spacings involved. Click on emphasised words in the text below
9124 to reveal the relation of image and code.}
9125 \mt@layer{_compatibility}{%
9126 \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
9127 \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
9128 \mt@pseudo@marg{\color{thered}%
9129 If you had a \acronym{PDF} viewer that understands

```

```

9130     \acronym{PDF}\, {\smaller1.5}, you could hide the arrows selectively.}}
9131     \vskip-\mt@unvdimen}%
9132     \vskip-4pt
9133     \setlength\fbboxsep{4pt}%
9134     \leavevmode
9135     \pdfstartlink
9136     user{/Subtype/Link
9137           /BS << /Type/Border/W 0 >> /H/0
9138           /A << /S/SetOCGState
9139             /State[/Toggle \mts@stuff] >>
9140           }%
9141     \fcolorbox{theframe}{theshade}%
9142     {\fontsize{34}{38}\selectfont #1}%
9143     \pdfendlink
9144     \par\medskip
9145     }%
9146     \def\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
9147 }

```

Now define the illustration to be used in the document.

```

9148 \def\lssample{%
9149   \ls@sample{%
9150     \dols{Opt}{Stop}
9151     \lssp{o}{0.45em}{0.25em}{0.15em}
9152     \dols{0.16em}{\st ealing}\hskip-\dimexpr 0.08em+\lsrule\relax
9153     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
9154     \dols{0.16em}{sheep}
9155     \dols{Opt}{!}
9156   }%

```

Don't forget to add the arrows.

```

9157   \vspace{-\baselineskip}
9158   \add@arrow{red}      {tracking}{\lsamount_c.east}{a_ls}
9159   \add@arrow{red}      {okern}   {okernend_c.east}{p_ls}
9160   \add@arrow{green}    {ospace}   {ospace_c.east}  {ospace}
9161   \add@arrow{green}    {ispace}   {ispace_c.center}{ispace}
9162   \add@arrow{green!75} {istretch}{istretch_c.east}{istretch.north}
9163   \add@arrow{green!75} {ishrink}  {ishrink_c.west} {ishrink.north}
9164   \add@arrow{green!75} {ostretch}{ostretch_c.east}{ostretch.north}
9165   \add@arrow{green!75} {oshrink}  {oshrink_c.east} {oshrink.north}
9166   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
9167 }
9168 \fi

```

This is for use with microtype.dtx

```

9169 \ifx\documentclass\@twoclasseserror
9170   \usepackage{tikz}
9171 \else

```

B.2 Document

```

9172 \documentclass[10pt,a4paper]{ltxdoc}
9173 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

```

Re-use the preamble from microtype.dtx.

```

9174 \usepackage{microtype-doc}
9175 \usepackage{attachfile}
9176 \usepackage{tikz}
9177 \makeatletter
9178 \pdfcatalog{/OCProperties <<
9179             /OCGs [\mt@objects]
9180             /D << /Order [\mt@order] /BaseState/OFF >>
9181             >> }
9182 \makeatother
9183 \begin{document}

```

You are currently reading this.

```

9184 \DocInput{microtype-lssample.dtx}
      Now show what we are able to do.
9185 \noindent
9186 Since a picture is worth a thousand words, probably even more if, in our
9187 case, it depicts a couple of letterspaced words, let's bring one to sum up
9188 these somewhat confusing options. Suppose you had the following settings
9189 (which I would in no way recommend; they are only for illustrative purposes):
9190 \begin{verbatim}
9191 \SetTracking
9192 [ no ligatures = {"\anchorarrow{nolig}"f},
9193   spacing      = {60"\anchorarrow{ispace}"0*,"%
9194                 "-1"\anchorarrow{istretch}"00*,""\anchorarrow{ishrink}"},
9195   outer spacing = {4"\anchorarrow{ospace}"50,"%
9196                 "2"\anchorarrow{ostretch}"50,1"\anchorarrow{oshrink}"50},
9197   outer kerning = {"\anchorarrow{okernbegin}"*,"%
9198                 "\anchorarrow{okernend}"*} ]
9199 { encoding = * }
9200 { 1"\anchorarrow{lsamount}"60 }
9201 \end{verbatim}
9202 and then write:
9203 \begin{verbatim}
9204 Stop \textls{stealing sheep}!
9205 \end{verbatim}
9206 this is the (typographically dubious) outcome:
9207
9208 \lssample
9209
9210 \noindent
9211 While the word 'Stop' is not letterspaced, the space between the letters in
9212 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
9213 of 160/1000\,em\,=\allowbreak\,0.16\,em.
9214 The \showarrow[ispace]{inner-space}{green} within the letterspaced text is
9215 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
9216 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
9217 untouched.
9218 The \showarrow[ospace]{outer-space}{green} (of 0.45\,em) immediately before the
9219 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
9220 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
9221 Note that there is no outer space after the text, since the exclamation mark
9222 immediately follows; instead, the default \showarrow[okern]{outer-kern}{red}
9223 of half the letterspace amount (0.08\,em) is added.
9224 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
9225 neglected to specify the '~|s|' in the |no ligatures| key.
9226
9227 \expandafter\enddocument
9228 \fi
9229 </lssample>

```

C Change history

2004/09/11 **Version 1.0**

General: Initial version 1

2004/09/21 **Version 1.1**

General: configuration file names in lowercase (suggested by <i>Harald Harders</i>)	82	<code>\MT@get@listname@</code> : don't check for empty attributes list	84
remove 8-bit characters from the configuration files (suggested by <i>Harald Harders</i>)	137	<code>\MT@ifempty</code> : fix: use category code 12 for the percent character (reported by <i>Tom Kink</i>)	43
Protrusion: add factors for some more characters	144	<code>\MT@is@number</code> : numbers may also be specified in hexadecimal or octal (suggested by <i>Harald Harders</i>)	89
settings for Adobe Minion (contributed by <i>Harald Harders</i>)	145	<code>\MT@pdfTeX@no</code> : fix: version check (reported by <i>Harald Harders</i>)	38
<code>\DeclareCharacterInheritance</code> : new command: possibility to specify character inheritance	112	<code>\MT@permute</code> : don't use sets for empty encoding	114
<code>\MT@declare@sets</code> : remove spaces around set name	99	<code>\MT@setup@expansion</code> : issue an error instead of a warning, when pdf \TeX version is too old for <code>autoexpand</code>	128
<code>\MT@DeclareSet</code> : remove spaces around first argument	98	<code>\MT@split@codes</code> : fix: allow zero and negative values	60
<code>\MT@find@file</code> : fix: also check whether the file for the base font family has already been loaded	82	<code>\MT@use@set</code> : remove spaces around set name	103
<code>\MT@get@basefamily</code> : only remove suffix if it is 'x' or 'j'	83	<code>\UseMicrotypeSet</code> : remove spaces around first argument	103

2004/10/03 **Version 1.2**

Font sets: declare <code>cmor</code> as an alias of <code>cmr</code>	135	<code>\MT@get@inh@list</code> : fix: set inheritance list <code>\globally</code> to <code>\empty</code>	86
new: <code>allmath</code> and <code>basimath</code>	134	<code>\MT@get@listname@</code> : alternatively check for alias font name	84
Protrusion: add settings for Computer Modern Roman and Adobe Garamond in T ₁ encoding	168	<code>\MT@get@size</code> : additional magic to catch some errors hijack <code>\set@fontsize</code> instead of <code>\setfontsize</code>	101
add settings for Computer Modern Roman math symbols	173	<code>\MT@loop</code> : fix: new macro, used instead of <code>\loop</code>	47
<code>\MT@familyalias</code> : define alias font name as an alternative, not as a replacement	56	<code>\MT@maybe@do</code> : also check for alias font name	57
<code>\MT@get@basefamily</code> : also remove 'w' (swash capitals)	83	<code>\MT@permute@@@@</code> : more sanity checks for <code>\SetProtrusion</code> and <code>\SetExpansion</code>	115
<code>\MT@get@highlevel</code> : check whether defaults have changed	99	<code>\MT@setupfont</code> : also search for alias font file	54
		fix: call <code>\@enc@update</code> if necessary	54

2004/10/27 **Version 1.3**

General: fix: specifying <code>load</code> option does no longer require to give a name, too	109	<code>\MT@fix@catcode</code> : check some category codes (compatibility with german)	34
Font sets: declare <code>aer</code> , <code>zer</code> and <code>hfor</code> as aliases of <code>cmr</code>	135	<code>\MT@load@list</code> : check whether list exists	82

2004/11/12 **Version 1.4**

General: check for <code>pdfcprot</code>	52	(OT1, T1, lmr)	150
don't use scratch registers in global definitions	86	<code>\microtypesetup</code> : fix: set the correct levels, and remember them; warning when enabling an option disabled in package options	123
use <code>\pickup@font</code> instead of <code>\define@newfont</code> as the hook for <code>\MT@setupfont</code>	93	<code>\SetExpansion</code> : fix: specifying extra options does no longer require to give a name, too	106
use one instead of five counters	48		
Protrusion: tweak quote characters for <code>cmr</code> variants			

2004/11/17 **Version 1.4a**

General: new option: <code>final</code>	120	when reading files (reported by <i>Michael Hoppe</i>)	83
<code>\MT@cfg@catcodes</code> : fix: reset some more catcodes			

2004/11/26 **Version 1.4b**

General: fix: set catcodes before reading global configuration file (reported by <i>Christoph Bier</i>)	122	form abczz (reported by <i>Georg Verweyen</i>)	83
optimisation: use less <code>\expandafters</code> and <code>\csnames</code>	42	<code>\MT@get@slot</code> : don't define <code>\MT@char</code> globally (save stack problem)	86
Protrusion: harmonise dashes in upshape and italic (<code>cmr</code> , <code>pad</code> , <code>pp1</code>)	144	<code>\MT@ifdimen</code> : don't set <code>\MT@count</code> globally (save stack problem)	44
slanted like italics	153	<code>\MT@setup@PDF</code> : new message if <code>\pdfoutput</code> is changed	126
<code>\MT@checklist@family</code> : fix: don't try alias family name if encoding failed	58	<code>\MT@use@set</code> : don't use undeclared font sets	103
<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 Thê Thành</i>)	120	<code>\MT@get@highlevel</code> : don't test defaults if called after begin document	99
new option: selected, by default false (suggested by <i>Hàn Thê Thành</i>)	118	<code>\MT@scale@factor</code> : warning for factors outside limits	62
Documentation: add 'Short history'	29	<code>\MT@scale@to@em</code> : don't use <code>\lcode</code> and <code>\rcode</code> for the calculation	61
add note about <code>DVIoutput</code> option	8	<code>\MT@set@ex@codes</code> : allow non-selected font expansion	66
Inheritance: remove <code>\ss</code> from T1 list, add <code>\DJ</code>	138	<code>\MT@set@pr@codes</code> : adjust protrusion factors before setting the inheriting characters	59
Protrusion: settings for Bitstream Charter	145	<code>\MT@setup@expansion</code> : defaults: calculate step as $\min(\text{stretch}, \text{shrink})/5$	127
<code>\DeclareMicrotypeAlias</code> : remove spaces around arguments	104	defaults: turn off expansion for DVI output	127
<code>\MT@cfg@catcodes</code> : reset catcode of '=' (compatibility with Turkish babel)	83	disable automatic expansion for DVI output	128
<code>\MT@fix@catcode</code> : reset catcode of '~' (compatibility with chemsym)	34		

2005/01/24 **Version 1.6**

General: defaults: turn off expansion for old pdfTeX versions	121	tune CMR math letters (OML encoding)	173
load a font if none is selected	54	<code>\MT@get@charwd</code> : use e-TeX's <code>\fontcharwd</code> , if available	61
new option: factor, by default 1000	120	<code>\MT@get@inh@list</code> : correct message if selected is false	86
restructure <code>dtx</code> file	134	<code>\MT@set@ex@codes</code> : introduce factor option	66
test whether <code>\pickup@font</code> has changed	96	<code>\MT@set@pr@codes</code> : introduce factor option	59
test whether numeric options receive a number	120	<code>\MT@setup@expansion</code> : disable automatic expansion for old pdfTeX versions	128
use e-TeX's <code>\ifcsm</code> and <code>\ifdefined</code> if defined	43	<code>\MT@use@set</code> : retain current set if new set is undeclared	103
Protrusion: add italic uppercase Greek letters	153	<code>\MT@vinfo</code> : new macro instead of <code>\ifMT@verbose</code>	35
improve settings for numbers (pointed out by <i>Peter Muthesius</i>)	146		

2005/02/02 **Version 1.6a**

Documentation: add table of fonts with tailored protrusion settings	21	reported by <i>Bernard Gaulle</i>)	86
<code>\MT@get@slot</code> : completely redone, hopefully more robust (compatible with <code>frenchpro</code> ; problem		<code>\MT@pdf@tex@no</code> : new macro	38
		<code>\MT@reset@ef@codes</code> : only reset <code>\efcodes</code> for older pdfTeX versions	67

2005/03/23 **Version 1.7**

General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i>)	100	<code>\textbackslashash</code> to T1 encoding	148
disallow automatic expansion if pdfTeX too old	112	<code>\DeclareMicrotypeAlias</code> : may also be used inside configuration files	104
fix: remove space after <code>autoexpand</code>	112	<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i>)	104
new value for verbose option: errors	120	<code>\Microtype@Hook</code> : new command for font package authors	122
shorter command names	48	<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	123
warning when running in draft mode	126	<code>\MT@begin@catcodes</code> : also use inside configuration	
Documentation: add hint about compatibility	26		
remove table of match order	12		
Protrusion: fix: remove <code>\</code> from OT1, add			

commands	83	sion	91
\MT@cfg@catcodes: reset catcode of ':' (compatibility with french* packages)	83	\MT@scale: new macro: use e-TeX's \numexpr if available	48
\MT@get@listname@: use \@tfor (<i>Andreas Bühmann's</i> idea)	84	\MT@set@ex@codes: two versions of this macro	66
\MT@get@slot: remove backslash hack	86	\MT@setup@expansion: modify \showhyphens	129
test for \chardefed commands	87	\MT@split@name: don't define \MT@encoding & \gloally	56
test whether \(\encoding)\(...) is defined	87	\MT@test@ast: make it simpler	100
\MT@if@list@exists: don't define \MT@#1@c@name \gloally, here and elsewhere	85	\MT@try@order: always check for size, too (suggested by <i>Andreas Bühmann</i>)	84
\MT@if@dimen: comparison with 1 to allow size smaller than 1 (suggested by <i>Andreas Bühmann</i>)	44	fix: also check for //(series)/\shape// (reported by <i>Andreas Bühmann</i>)	84
\MT@increment: use e-TeX's \numexpr if available	48	\MT@warn@code@too@large: new macro: type out maximum protrusion factor	63
\MT@is@composite: new macro: construct command for composite character; no uncontrolled expansion		\MT@warn@err: new macro: for verbose=errors	35

2005/06/23 **Version 1.8**

General: \SetProtrusion: new key: unit	111	\MT@find@file: no longer wrap names in commands	82
if font substitution has occurred, set up the substitute font, not the selected one	94	\MT@get@charwd: warning for missing (resp. zero-width) characters	61
new option: config to load a different main configuration file	122	\MT@get@font@dimen@six: new macro: test whether \fontdimen 6 is defined	59
new option: unit, by default character	121	\MT@get@listname@: made recursive	84
Documentation: add example for factor option	13	\MT@get@slot: fix: expand active characters	86
add example of how to get rid of a widow (suggested by <i>Adam Kucharczyk</i>)	14	test whether \(\encoding)\(...) is defined made more robust	87
add hint about error messages	27	\MT@get@unit: new macro: get unit for codes	64
Font sets: add U encoding to allmath	134	\MT@in@rlist: made recursive	47
declare pxr and txr as aliases of ppl resp. ptm	135	\MT@is@active: new macro: translate inputenc-defined characters	90
Inheritance: remove \DJ from T1 list (it's the same as \DH)	138	\MT@is@letter: warning for non-ASCII characters	89
Protrusion: add LY1 characters for Times	153	\MT@ledmac@setup: character protrusion with ledmac	50
settings for AMS math fonts	177	\MT@map@clist@n: new macro: used instead of \@for	46
verified settings for slanted Computer Modern Roman	161	\MT@map@tlist@n: new macro: used instead of \@tfor	46
\add@accent: fix: disable micro-typographic setup inside \add@accent (reported by <i>Stephan Hennig</i>)	95	\MT@old@cmd: renamed commands from \..MicroType.. to \..Microtype..	35
\DeclareMicrotypeAlias: warning when overriding an alias font	104	\MT@pdf@tex@no: case 5: pdfTeX 1.30	38
\DeclareMicrotypeSetDefault: new command: set default font set	103	\MT@permute@@@@@: add ranges to the beginning of the lists	115
\MT@cfg@catcodes: reset catcodes of the remaining ASCII characters	83	\MT@scale: fix: remove spaces in e-TeX variant (reported by <i>Mark Rossi</i>)	48
\MT@check@rlist: made recursive	116	\MT@setup@font@hook: restore \% and \# when hyperref is loaded	51
\MT@curr@list@name: new macro: current list type and name	92	restore csquotes's active characters	51
\MT@declare@sets: warning when redefining a set	99	restore percent character if Spanish babel is loaded	51
\MT@define@set@key@: use comma lists instead of token lists	99	\MT@split@codes: get character width once only	60
		\MT@use@set: fix: remove braces in first line	103
		\MT@xadd: simplified	46

2005/10/28 **Version 1.9**

General: \DeclareMicrotypeSet: new key: font	101	Documentation: add hint about verbatim environment	25
\SetProtrusion: value 'relative' renamed to 'character' for key unit	111	add remark about Type 1 fonts required for automatic font expansion	8
allow context-specific font setup	94	Font sets: add OT4 encoding to text sets	134
compatibility with TeX Live hack (reported by <i>Herbert Voß</i>)	37	add T5 encoding to text sets	134
disable microtype setup inside hyperref's \pdfstringdef (reported by <i>Hàn Thế Thành</i>)	52	declare qpl and qtm (qfonts, TeX Gyre) as aliases of ppl resp. ptm	135
fix: use true as the default value	118	Inheritance: add list for OT4	139
option unit: rename value relative to character	121	add list for T5 (requested by <i>Hàn Thế Thành</i>)	140

Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR	148	<code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists	64
settings for OT4 encoding (Computer Modern Roman, Palatino, Times)	144	<code>\MT@is@active</code> : redone: use <code>\set@display@protect</code>	90
settings for T5 encoded Computer Modern Roman	144	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code>	89
<code>\DisableLigatures</code> : new command: disable ligatures (requires pdf \TeX 1.30)	105	<code>\MT@maybe@do</code> : redone	57
<code>\microtypecontext</code> : new command: change setup context in the document	97	<code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list	47
<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code>	58	<code>\MT@scale@factor</code> : generalised	62
<code>\MT@detokenize@c</code> : fix the \TeX version	43	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero	129
<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code>	43	warning if user requested zero step	127
		<code>\MT@toks</code> : use instead of <code>\toks@</code>	40
		<code>\SetProtrusion</code> : (et al.) new key: font	105

2005/12/05 **Version 1.9a**

General: ‘ <i>file name</i> ’/‘ <i>line number</i> ’ as default list name	109	diately (requested by <i>Georg Verweyen</i>)	99
new option: <code>defersetup</code> , by default true	119	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed	99
remove superfluous test whether <code>\pickup@font</code> has changed	96	<code>\MT@ifdefined@c@T</code> : new macros: true case only	43
Documentation: add explanation for error message in DVI mode	27	<code>\MT@ifint</code> : use <code>\pdfmatch</code> if available	44
add explanation for error message with non-Type 1 fonts	27	<code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available	45
Font sets: declare <code>mbch</code> (<code>mathdesign</code>) as an alias of Charter	136	<code>\MT@in@clist</code> : fix	46
Protrusion: fix: remove ‘_’ from OT1 encoding	149	<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)	62
settings for T5 encoded Charter	144	<code>\MT@is@feature</code> : new macro: check for pdf \TeX feature	49
<code>\microtypesetup</code> : inside the preamble, accepts all package options	123	<code>\MT@map@clist@n</code> : following \LaTeX 3	46
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	96	<code>\MT@permute@t@t@t</code> : don’t define permutations for unused encodings	115
<code>\MT@define@set@key@</code> : don’t expand variables immediately		<code>\MT@rem@from@clist</code> : fix	47
		<code>\MT@setup@</code> : defer setup until the end of the preamble	49

2006/01/20 **Version 1.9b**

General: compatibility with listings: sanitise more catcodes (reported by <i>Holger Uhr</i>)	53	add samples of micro-typographic features	4
compatibility with the <code>extendedchar</code> option of the listings package	53	<code>\MT@features</code> : use throughout the package to adjust to beta-ness	49
Documentation: activate expansion in the distributed PDF	1	<code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available	44
		<code>\MT@warn@code@too@large</code> : fix calculation with present factor	63

2006/02/02 **Version 1.9c**

Documentation: add example of how to increase protrusion of footnote markers (suggested by <i>Georg Verweyen</i>)	22	<code>\MT@define@code@key@font</code> : fix: context was ignored	108
Protrusion: settings for URW Garamond	145	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@temp@size</code> in <code>\csname</code> (bug introduced in v1.9b)	108

2006/05/05 **Version 1.9d**

Font sets: <code>md*</code> instead of <code>m</code> series in basic sets	134	<code>\DeclareCharacterInheritance</code> : fix: empty context	112
add QX encoding to text sets	134	<code>\MT@detokenize@n</code> : new macro: use <code>\detokenize</code> if available	43
Inheritance: add list for QX encoding (contributed by <i>Maciej Eder</i>)	139	<code>\MT@get@ex@opt</code> : fix: evaluate preset	67
Protrusion: settings for QX encoding (contributed by <i>Maciej Eder</i>)	151	<code>\MT@get@font@dimen</code> : warning for zero fontdimen	62
settings for Euro symbols (Adobe, ITC, marvosym)	184	<code>\MT@get@opt</code> : optimise: don’t reset when preset option is set	64
tweak AMS settings	177	set list name before presetting	64

<code>\MT@is@active</code> : support for Unicode (inputenc/utf8)	90	mandatory argument	105
<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when tex4ht is loaded (reported by <i>Peter Dyballa</i>)	51	(et al.) split keys of optional and mandatory argument	105
<code>\SetProtrusion</code> : (et al.) optimise: unify keys for			

2006/07/28 **Version 1.9e**

General: fix: default value for <code>activate</code> : true	118	settings for Euler Roman font	180
Documentation: add hint about unknown encodings include LPL	26 238	<code>\DeclareCharacterInheritance</code> : new key ‘inputenc’ to set the input encoding	112
Font sets: declare <code>zeur</code> and <code>zeus</code> (eulervm) as aliases of <code>eur</code> resp. <code>eus</code> (euler)	136	<code>\MT@rem@from@clist</code> : model after <code>\@removeelement</code>	47
Inheritance: adapt to marvosym’s changed encoding	141	<code>\MT@setup@</code> : empty <code>\MT@setup@</code> after use (compatibility with the combine class)	49
Protrusion: complete settings for Euler Fraktur and Script fonts	183	<code>\pickup@font</code> : no tracing with trace package	95
fix: forgotten comma in <code>mt-mvs.cfg</code> ; adapt to marvosym’s changed encoding	184	<code>\SetExpansion</code> : new key: <code>inputenc</code>	106
		<code>\SetProtrusion</code> : (et al.) new key: <code>inputenc</code>	105

2006/09/09 **Version 1.9f**

Protrusion: fix: <code>euler-vm</code> did not load <code>euler</code> settings	181	<code>\MT@reset@context</code> : only reset context if it has actually been changed	97
<code>\MT@curr@list@name</code> : fix: <code>\MessageBreak</code> must not be expanded	92	<code>\MT@set@inh@list</code> : fix: forgotten comma in the features list	113
<code>\MT@gdef@n</code> : new macros: global variants	42	<code>\MT@set@named@keys</code> : new macro: set name first, simplify parsing of optional argument	107
<code>\MT@get@inh@list</code> : fix: input encoding must be set after the inheritance list has been parsed	86	<code>\SetProtrusion</code> : (et al.) set catcodes before parsing optional argument	105
<code>\MT@glet</code> : new macro	42		

2007/01/14 **Version 2.0**

General: (beta:1) new option: <code>babel</code> , by default false (language-dependent setup suggested by <i>Ulrich Dirr</i>)	118	Font sets: add <code>footnotesize</code> and <code>scriptsize</code> sets	135
(beta:1) new option: <code>letterspace</code> , by default 100	120	add <code>smallcaps</code> set	134
(beta:3) compatibility with <code>soul</code> : register <code>\textls</code> and <code>\sstyle</code>	53	<code>\DeclareMicrotypeBabelHook</code> : (beta:1) new command: interaction with <code>babel</code>	105
(beta:8) option ‘ <code>babel</code> ’: fix: switch off French <code>babel</code> ’s shorthands properly (reported by <i>Daniel Flipo</i>)	133	<code>\sstyle</code> : (beta:1) new command: <code>letterspacing</code>	75
(beta:8) option ‘ <code>babel</code> ’: switch off Turkish <code>babel</code> ’s shorthands	133	(beta:8) fix: font switches don’t pose a problem anymore	75
compatibility with <code>listings</code> : set catcode of backslash to zero (reported by <i>Steven Bath</i>)	53	(beta:8) fix: <code>letterspacing</code> commands may be nested	75
new package <code>letterspace</code> : a stripped-down version, containing the <code>letterspacing</code> commands only	1	totally redone, using the new <code>\letterspacefont</code>	75
option ‘ <code>unit</code> ’, <code>\SetProtrusion</code> : deprecate value ‘ <code>relative</code> ’ completely	111	<code>\MT@declare@sets</code> : fix: empty size list when redefining set	99
Documentation: add hint about how to increase <code>font_max</code> and <code>font_mem_size</code>	27	<code>\MT@is@symbol</code> : made even more robust	91
add hint about warning when tracking <i>and</i> expansion is applied to a font	27	<code>\MT@load@inputenc</code> : sanitise catcodes before loading input encoding (problem with <code>listings</code>)	65
add remark about ‘ <code>draft</code> ’ option disabling microtype (noted by <i>Michalis Miatidis</i>)	9	<code>\MT@pdftex@no</code> : (beta:1) case 6: pdfTeX 1.40	38
qualify hint about web documents with regard to older pdfTeX versions	25	<code>\MT@setup@noligatures</code> : maybe disable <code>\MT@noligatures</code> after the preamble	132
qualify hints about expansion error messages with regard to older pdfTeX versions	27	<code>\MT@split@name</code> : adjust to possible <code>letterspacing</code>	56
		<code>\SetExtraKerning</code> : (beta:1) new command: additional kerning	107
		<code>\SetExtraSpacing</code> : (beta:1) new command: adjustment of interword spacing	107
		<code>\SetTracking</code> : new command: tracking	106
		<code>\textls</code> : (beta:1) new command: <code>letterspacing</code>	79
		starred version: remove spaces around text	79
		<code>\tracingmicrotypeinpdf</code> : new debug method: mark all fonts with PDF annotations	36

2007/01/21 **Version 2.1**

General: compatibility with CJK: also check for its definition	94	<code>\lslig</code> : new command: protect ligatures in letter-spaced text	75
compatibility with pinyin: disable microtype in <code>\py@macron</code> (reported by <i>Sven Naumann</i>)	53	<code>\MT@get@ls@basefont</code> : redone: use <code>\pdfmatch</code> to make it bullet-proof	75
fix: letterspace package forgot to load <code>keyval</code>	40	<code>\textls</code> : fix: use <code>\hmode@bgroup</code>	79

2007/07/14 **Version 2.2**

General: disable microtype if wordcount is loaded (reported by <i>Ross Hetherington</i>)	49	<code>\MT@is@symbol</code> : expand once more (for frenchpro)	91
new option: <code>copyfonts</code>	119	<code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code>	72
simplify key declarations	109	<code>\MT@lua</code> : (basic) support for Lua \TeX	39
Documentation: add hint about error message with pdf \TeX 1.40	27	<code>\MT@maybe@etex</code> : use catcode trickery	37
add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i>)	25	<code>\MT@pdf@tex@no</code> : case 7: pdf \TeX 1.40.4	38
add overview	5	<code>\MT@preset@aux@space</code> : generalised	65
logo transparency and amusement	1	<code>\MT@set@all@pr</code> : (et al.) allow empty values	60
Font sets: declare FPL Neu as an alias of Palatino	136	<code>\MT@set@inputenc@</code> : only load inputenc files if necessary	64
declare <code>chr</code> (<code>chmath</code>) as an alias of Charter (reported by <i>Geoff Vallis</i>)	136	<code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in pdf \TeX 1.40.4)	72
default set for tracking: <code>smallcaps</code>	135	possibility to customise interword spacing	72
Inheritance: remove ‘-’ \rightarrow ‘127’	138	<code>\MT@setup@expansion</code> : warning if stretch or shrink aren’t multiples of step	129
Protrusion: settings for Bitstream Letter Gothic	145	<code>\MT@setup@font</code> : don’t call <code>\@@enc@update</code> anymore	54
Spacing: add sample	185	only add features that are available with the respective pdf \TeX	54
Tracking: add ligatures that are to be disabled	142	<code>\MT@setup@font@hook</code> : restore percent character if Galician <code>babel</code> is loaded	51
<code>\DeclareMicrotypeVariants</code> : new command	104	<code>\MT@the@pr@code@tr</code> : adjust protrusion of letter-spaced fonts	60
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only	105	<code>\MT@tracking</code> : remember fonts that shouldn’t be letterspaced	71
<code>\lslig</code> : always defined	75	<code>\MT@tracking@</code> : fix: tracking couldn’t be re-enabled	71
<code>\MT@checklist@font</code> : fix: construct font name from characteristics	58	<code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode	131
<code>\MT@copy@font</code> : optionally work on copies of fonts	55	<code>\MT@with@babel@and@T</code> : also inspect class options	50
<code>\MT@get@basefamily</code> : redone, working on font names and suffixes of arbitrary length	83	<code>\pickup@font</code> : letterspace: setup inside group	95
<code>\MT@get@charwd</code> : subtract letterspacing amount from width	62	<code>\SetTracking</code> : new key ‘no ligatures’ to disable ligatures of letterspaced fonts	106
<code>\MT@get@ls@basefont</code> : fix again: remember base font in a macro	75	new keys ‘spacing’ and ‘outer spacing’ to adjust interword spacing (suggested by <i>Steven E. Harris</i>)	106
<code>\MT@ifdimen</code> : employ Lua \TeX features if available	44	third argument may be empty	106
<code>\MT@ifint</code> : employ Lua \TeX features if available	44	<code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code>	97
<code>\MT@ifstreq</code> : employ Lua \TeX features if available	45		
fix: ϕ - \TeX version shouldn’t use <code>\x</code> and <code>\y</code> (found by <i>Wiebke Petersen</i>)	45		
<code>\MT@is@composite</code> : more robust: expand exactly once	91		

2007/12/23 **Version 2.3**

General: disable <code>\microtypecontext</code> in <code>hyperref</code> ’s <code>\pdfstringdef</code>	52	<i>phan Hennig</i>)	97
fix: really switch off Turkish shorthands	133	<code>\MT@begin@catcodes</code> : fix: don’t disable <code>\KV@sp@def</code>	83
new value for verbose option: <code>silent</code> (suggested by <i>Karl Berry</i>)	120	<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	101
turned some warnings into errors	120	<code>\MT@in@clist</code> : fix: don’t use <code>\x</code> (reported by <i>Peter Meier</i>)	46
Documentation: add kerning sample	18	<code>\MT@is@active</code> : support for extended Unicode (inputenc/utf8x resp. ucs) – experimental	90
add letterspacing illustration	17	<code>\MT@noligatures</code> : fix: set evaluation didn’t work (bug introduced in v2.2)	81
<code>\do@subst@correction</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i>)	95	<code>\MT@plain</code> : letterspace: support for <code>eplain/miniltx</code>	37
<code>\lslig</code> : redone: extract outer kerns from current letterspacing amount	75	<code>\MT@set@curr@os</code> : adjusting spaces made more reliable	73
<code>\microtypecontext</code> : made robust (reported by <i>Ste-</i>			

<code>\MT@set@tr@codes</code> : also adjust tracking if protrusion is not enabled, and even for letterspace (reported by <i>Stephan Hennig</i>)	72	<code>\MT@SetTracking</code> : sanity check for value	107
possibility to customise outer kerning (suggested by <i>Stephan Hennig</i>)	72	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled	130
		<code>\MT@tr@outer@l</code> : only change pre outer space if it contains shrink	77

2008/02/29 **Version 2.3a**

General: fix test for <code>soul</code> under plain \TeX	53	<code>\MT@fix@catcode</code> : fix catcodes earlier, and also for the letterspace package	34
Documentation: add hint about <code>babel</code> having to be loaded first	26	<code>\MT@getkey</code> : fix: <code>key=val</code> in class options list	126
add table of available and enabled features	7	<code>\MT@maybe@etex</code> : removed	37
mention <code>soulutf8</code>	29	<code>\MT@set@codes</code> : generalised	60
Protrusion: adjust LMR quotation marks again	150	<code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded	51
<code>\MT@error@doesnt@work</code> : error messages if <code>pdfTeX</code> is too old for extensions	131		

2008/06/04 **Version 2.3b**

General: compatibility with <code>CJKutf8</code> : also check for its definition	94	<code>\MT@requires@latex</code> : new macro	37
<code>\MT@exp@gcs</code> : new macro: reduce save stack size	42	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i>)	72
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i>)	55	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode	77
<code>\MT@get@size</code> : grouping	101	make <code>\spaceskip-aware</code> (<code>ragged2e</code>)	77
<code>\MT@noligatures@</code> : fix: warning messages for unknown slots	81	<code>\MT@tr@outer@r@@</code> : additional test for horizontal mode	78

2008/11/11 **Version 2.3c**

General: Lua \TeX supported by default	39	coding (reported by <i>Vasile Gaburici</i>)	140
Documentation: add hint about spacing being experimental	26	<code>\MT@detokenize@c</code> : fix: remove last space only (reported by <i>Ulrich Durr</i>)	43
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>)	78
Inheritance: add <code>\textcommabelow[STst]</code> to QX encoding			

2009/03/27 **Version 2.3d**

General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i>)	53	(reported by <i>Ulrich Durr</i>)	75
move setup to the very end (for <i>Colin Rourke</i>)	134	<code>\MT@setup@expansion</code> : default step: 1 for <code>pdfTeX</code> versions ≥ 1.40	127
<code>\ifMT@inannot</code> : use <code>pdfTeXcmds</code> for debugging	36	<code>\MT@tr@outer@r@@</code> : don't use <code>\x</code> (reported by <i>Ulrich Durr</i>)	78
<code>\lststyle</code> : disable for Lua \TeX	75	fix: don't adjust in math mode (reported by <i>Christoph Bier</i>)	78
make invalid in math mode	75	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i>)	78
<code>\microtypesetup</code> : select font after setup	123	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning	80
<code>\MT@check@active@set</code> : warning for missing default sets	122	<code>\textls</code> : make math mode aware	79
<code>\MT@lua</code> : update for Lua \TeX 0.36	39		
<code>\MT@set@tr@codes</code> : allow zero tracking	72		
<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>	25	<i>Karl Karlsson</i>)	187
Expansion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	142	<code>\MT@get@font@dimen@six</code> : fix: gobbling settings with tracking failed (reported by <i>Leo</i>)	59
Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set	134	<code>\MT@setup@</code> : make space-unaware (requested by <i>Marcin Borkowski</i>)	49
add T2A encoding	134	<code>\MT@tr@outer@r</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i>)	77
Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	150	<code>\MT@tr@outer@r@@</code> : fix: set current kerning and spacing again (found by <i>Lars Rönmbäck</i>)	78
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*) 42

Protrusion: settings for T2A encoded Minion (contributed by *Karl Karlsson*) 150

2013/03/13 **Version 2.5**

General: allow contexts for LuaTeX 109

disable ‘DVIoutput’ option for XeTeX 119

fix: check whether ‘*file*’/‘*line*’ list name already exists (reported by *Till A. Heilmann*) 109

letterspacing with LuaTeX 0.62 71

new files: `microtype-pdftex.def`, `microtype-xetex.def`, `microtype-luatex.def`, containing engine-specific definitions 37

protrusion with XeTeX 39

restore `\space` inside listings (reported by *Rolf Dieterich*) 53

Documentation: add hint about spacing and `ragged2e` 26

add hint about `dtx` source code 27

add hint about LuaTeX compatibility 26

include `microtype-logo.dtx` and `microtype-ls-sample.dtx` 212

Font sets: add EU1 and EU2 encodings 134

declare `lmr` as alias of Latin Modern Roman (OpenType version) 135

declare `lmsy` and `lmm` as aliases of `cmsy` resp. `cmm` (reported by *Jonas Hogstrom*) 135

declare `zgmx` etc. (`garamondx`) as aliases of URW Garamond 136

declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of `Palatino Linotype` (OpenType version) 136

Inheritance: add rudimentary list for EU1 and EU2 141

Protrusion: add default lists for EU1 and EU2 149

improvements to Computer Modern Roman italics (contributed by *Hendrik Vogt*) 153

Tracking: add EU2 encoding to default list 142

`\DeclareCharacterInheritance`: allow more than one encoding 112

`\DeclareMicrotypeAlias`: remove spaces in font name 104

`\ifMT@nofamily`: info if settings are not family-specific (suggested by *Hàn Thê Thành*) 59

`\LoadMicrotypeFile`: remove all spaces in font name 104

`\lsstyle`: fix: ensure to set up math fonts (reported by *RazorXsr*) 75

`\microtypecontext`: fix: ensure to set up math fonts (reported by *RazorXsr*) 97

`\MT@define@code@key@family`: compatibility with `fontspec`: remove its internal counter (reported by *Till A. Heilmann*) 108

`\MT@define@code@key@font`: scrub `fontspec` feature count (found by *Meho R*) 108

`\MT@do@font`: adapt for XeTeX 48

adapt for LuaTeX 48

`\MT@get@slot@`: adapt for XeTeX 87

adapt for LuaTeX (requested by *Georg Duffner*) 87

`\MT@if@outer@next`: fix: conflict with `amsmath` (reported by *Scott Pakin*) 78

`\MT@info@missing@char`: fix error message for XeTeX (reported by *Juan Acevedo*) 62

`\MT@is@charx`: compatibility with `xunicode` 91

`\MT@ledmac@setup`: fix to work with XeTeX (reported by *Maieul Rouquette*) 50

`\MT@ls@set@ls`: allow formulas in optional argument to `\textls` (fix by *Heiko Oberdiek*) 80

`\MT@register@subst@font`: only register if it isn’t registered already (reported by *George Gratzner* and *Josep Maria Font*) 96

`\MT@register@subst@font@cx`: only register if it isn’t registered already 96

`\MT@scrubfeatures`: compatibility with `fontspec`: remove its internal counter 56

`\MT@set@all@pr`: fix: remove space (found by *Meho R*) 60

`\MT@set@pr@codes`: make info about generic settings encoding-specific (reported by *Sebastian Schuber*) 59

`\MT@setup@spacing`: warning with `ragged2e` (reported by *Steffen Hoffmann*) 130

`\MT@setupfont@hook`: restore `\%` and `\#` when `mathastext` is loaded (found by *Seamus Bradley*) 51

select font with `fontspec` (found by *Georg Duffner*) 51

2013/05/23 **Version 2.5a**

General: use `luatexbase` instead of `luatextra` (contributed by *Élie Roux*) 41

Documentation: add hint about `fontspec` having to be loaded first 26

include OpenType configuration files 189

`\MT@afteraftergroup`: fix: get outer kerning and spacing of nested `letterspacing right` 74

`\MT@get@slot@`: adapted for `luaotfload v2.2` (contributed by *Élie Roux*) 88

`\MT@led@unhbox@line`: simplified 50

`\MT@ledmac@setup`: support for `eledmac` 50

`\MT@ls@outer@k`: add marker for tightly nested `letterspacing` 81

`\MT@set@tr@codes`: fix: load font for `fontspec` 73

`\MT@xspace`: fix outer spacing problem with `xspace` (reported by *Dave*) 79

D Index

Links (in blue) refer to the page where the corresponding entry is described (bold face) resp. occurs. Plain numbers (in black) refer to the code line where the corresponding entry is defined (underlined) resp. used.

- Options:**
- DVIoutput **8**
 - activate **6**
 - auto **7**
 - babel **9**
 - config **9**
 - draft **9**
 - expansion **6**
 - factor **7**
 - final **9**
 - kerning **6**
 - letterspace **8**
 - protrusion **6**
 - selected **8**
 - shrink **8**
 - spacing **6**
 - step **8**
 - stretch **8**
 - tracking **6**
 - unit **7**
 - verbose **9**
- User Commands:**
- \DeclareCharacterInheritance **20**
 - \DeclareMicrotypeAlias **21**
 - \DeclareMicrotypeBabelHook **23**
 - \DeclareMicrotypeSet **10**
 - \DeclareMicrotypeSet* **10**
 - \DeclareMicrotypeSetDefault **12**
 - \DeclareMicrotypeVariants **20**
 - \DeclareMicrotypeVariants* **20**
 - \DisableLigatures **24**
 - \LoadMicrotypeFile **21**
 - \lslig **23**
 - \lssstyle **23**
 - \microtypecontext **22**
 - \microtypesetup **9**
 - \SetExpansion **14**
 - \SetExtraKerning **18**
 - \SetExtraSpacing **19**
 - \SetProtrusion **12**
 - \SetTracking **15**
 - \textls **23**
 - \textls* **23**
 - \textmicrotypecontext **22**
 - \UseMicrotypeSet **11**
- A**
- a0poster (package) **101**
 - activate (option) **6, 118, 226**
 - \add@accent **2735**
 - ae (package) **21, 135**
- B**
- babel (option) **9, 23, 26, 31, 118, 226**
- C**
- chemsym (package) **223**
 - chmath (package) **21, 136, 227**
 - CJK (package) **27, 53, 94, 227, 228**
 - CJKutf8 (package) **94, 228**
 - cm-super (package) **8**
 - color (package) **9, 126**
 - combine (package) **134, 226**
- D**
- \DeclareCharacterInheritance **20, 41, 3522**
 - \DeclareMicrotypeAlias **21, 43, 71, 3120**
 - \DeclareMicrotypeBabelHook **23, 45, 64, 3182**
 - \DeclareMicrotypeSet **10, 32, 57, 59, 72, 2848**
 - \DeclareMicrotypeSet* **10, 2848**
 - \DeclareMicrotypeSetDefault **12, 34, 61, 3079**
 - \DeclareMicrotypeVariants **20, 42, 58, 63, 3107**
 - \DeclareMicrotypeVariants* **20**
- E**
- eco (package) **21, 135**
 - \efcode **1455, 1481, 1482, 1522, 1524**
 - eledmac (package) **50, 229**
 - eplain (package) **24, 30, 33, 37, 50, 126, 227**
 - eucal (package) **181**
 - eufrak (package) **183**
- F**
- \f@family **2689, 2690**
 - \f@size **2682, 2684, 2686, 2693, 2733**
 - factor (option) **7, 13, 32, 120, 121, 223**
 - fancyvrb (package) **26**
 - final (option) **9, 32, 118, 120, 222**
 - fix-cm (package) **56**
 - \font **898, 1843**
 - \font@name **111, 943, 945, 947, 949, 959, 960, 1714, 1719, 1754, 1760, 1765, 1768, 1829, 1843, 1855, 1857, 1862, 1864, 1867, 1870, 1873, 1908, 1963, 2105, 2107, 2675, 2683, 2692, 2695, 2716, 2718, 2720, 2734, 2747, 2748, 2765, 2770**
 - \fontcharwd **1199, 1203, 1209**
- amsmath (package) 229**
- amssymb (package) 177**
- article (package) 22**
- auto (option) 7, 8, 118**
- babel (package) 2, 3, 5, 16, 23, 24, 26, 51, 83, 105, 132, 133, 136, 223, 224, 226–228**
- config (option) 9, 20, 31, 122, 224**
- contour (package) 126**
- copyfonts (option) 55, 118, 119, 227**
- crop (package) 126**
- csquotes (package) 28, 51, 148, 224**
- \curr@fontshape 2682, 2684, 2686, 2693, 2733**
- deferssetup (option) 51, 118, 119, 123, 225**
- \define@newfont 2675, 2683, 2687, 2695**
- \DisableLigatures 24, 40, 62, 3155**
- \do@subst@correction 2732**
- docstrip (package) 33**
- draft (option) 9, 118, 120, 226**
- dsfont (package) 59**
- DVIoutput (option) 8, 9, 118, 119, 223, 229**
- euler (package) 92, 180, 226**
- eulervm (package) 21, 136, 181, 226**
- euroitc (package) 184**
- europs (package) 184**
- eurosans (package) 184**
- expansion (option) 6, 11, 118, 127**

- `\fontdimen` 1122,
 1129, 1234, 1239, 1338, 1341, 1731, 1732,
 1760, 1855, 1901, 1902, 1909, 1910, 1916,
 1920, 1927, 1937, 1941, 1944, 1963, 2118, 2121
`fontinst` (package) 188, 212, 214
`fontinstallationguide` (package) 167
- G**

`garamondx` (package) 21, 136, 229
`german` (package) 34, 222
- H**

`hfoldsty` (package) 21, 135
- I**

`IEEEtran` (package) 94
`\iffontchar` 1244
`\ifMT@auto` 282, 4253, 4304
`\ifMT@babel` 282, 4380, 4504
`\ifMT@do` 992, 1055, 1690, 2152
`\ifMT@document` 323, 3633
`\ifMT@draft` 282, 4158
`\ifMT@expansion` 282, 4216, 4301, 4337
`\ifMT@fontspec`
 .. 782, 787, 816, 973, 1750, 2912, 3306, 3332
`\ifMT@if@` 281,
 792, 799, 823, 855, 2761, 3736, 4524, 4527
`\ifMT@inannot` 99
`\ifMT@inlist@` ... 597, 642, 715, 740, 916, 1022,
 1039, 1048, 1066, 1084, 1688, 2205, 2216,
 2265, 2314, 2748, 2755, 2767, 2822, 2834, 3138
`\ifMT@kerning` 282, 4394, 4517
- J**

`jurabib` (package) 94
- K**

`kerning` (option) 6, 11, 30, 118
`keyval` (package) 40, 60, 137, 227
`\knaccode` 1639, 1640, 1650, 1653, 1659
- L**

`ledmac` (package) 28, 31, 50, 94, 224
`ledpar` (package) 28
`\leftmarginkern` 753
`letterspace` (option) 8, 17, 23, 24, 30, 44, 71, 120, 226
`letterspace` (package) 1,
 24, 30, 31, 33, 35, 37, 72, 73, 75, 226–228
`\letterspacefont` 1719
`lineno` (package) 50
`listings` (package) 26, 53, 65, 83, 225, 226, 229
- M**

`marvosym` (package) ... 21, 31, 33, 141, 184, 225, 226
`mathastext` (package) 51, 229
`mathdesign` (package) 21, 136, 225
`memoir` (package) 22, 94
`\Microtype@Hook` 122, 3998
`\microtypecontext`
 .. 22, 47, 844, 2791, 2802, 4165, 4487, 4490
`microtypecontext` (environment) 22
`\microtypesetup` 9, 46, 4002, 4164
`miniltx` (package) 24, 30, 33, 37, 50, 227
`minimal` (package) 54
`\MT@font` 380, 919,
 941, 1124, 1235, 1248, 1702, 2268, 2340, 2721
`\MT@abbr@ex` 691
`\MT@abbr@ex@c` 691
`\MT@abbr@ex@inh` 691
`\MT@abbr@kn` 691
`\MT@abbr@kn@c` 691
`\MT@abbr@kn@inh` 691
`\MT@abbr@nl` 691
`\MT@abbr@pr` 691
`\MT@abbr@pr@c` 691
`\MT@abbr@pr@inh` 691
`fontspec` (package) 6,
 22, 24, 26, 27, 29, 30, 56, 100, 108, 135, 229
`fourier` (package) 59
`french` (package) 224
`frenchpro` (package) 91, 223, 227
- `\gls@currsiz` 1832, 2796
`graphics` (package) 9, 126
`hyperref` (package) 9, 51, 52, 94, 126, 224, 227
`\ifMT@nofamily` 1094, 1101
`\ifMT@noligatures` 282, 4470
`\ifMT@nonselected` 1432, 2327, 2335
`\ifMT@noreset` 2396, 2413, 2425, 2439, 2500
`\ifMT@opt@auto` 3743, 4256
`\ifMT@opt@DVI` 3743, 4183
`\ifMT@opt@expansion` 3743, 4212, 4338
`\ifMT@protrusion` 282, 747, 769, 4195, 4353
`\ifMT@selected` 282, 4287, 4307
`\ifMT@spacing` 282, 4362, 4379
`\ifMT@tracking` 282, 4349, 4409
`\ifMT@xunicode` 782, 2583, 2621
`ifpdf` (package) 9, 94, 126
`\iftracingmicrotypeinpdfall` 105
`inputenc` (package) 14,
 26, 31, 53, 64, 65, 86, 90, 92, 224, 226, 227
- `\knbcbcode` 1632, 1633, 1649, 1652, 1658
`\knbscode` 1553, 1554, 1577, 1581, 1587
- `lmodern` (package) 8, 150
`\LoadMicrotypeFile` 21, 44, 74, 3134, 4163
`\lpcode` 1136, 1172, 1173, 1376, 1379, 1744
`\lslig` 23, 54, 1848, 1851
`\lssstyle` 23, 51, 846, 871, 876,
 1830, 2044, 2050, 2055, 2058, 2068, 2072, 4166
`luaotfload` (package) 26, 88, 229
`luatexbase` (package) 41, 229
`luaextra` (package) 229
- `\MT@abbr@sp` 691
`\MT@abbr@sp@c` 691
`\MT@abbr@sp@inh` 691
`\MT@abbr@tr` 691
`\MT@abbr@tr@c` 691
`\MT@active@features`
 951, 2749, 2752, 2764, 2775, 2811, 2821,
 3163, 4196, 4302, 4350, 4363, 4395, 4479, 4480
`\MT@addto@annot` 99
`\MT@addto@setup` 731, 732,
 802, 1351, 1976, 2669, 2791, 2792, 3393,
 3411, 3856, 4003, 4157, 4440, 4478, 4503, 4536
`\MT@adjustspacing` 4063, 4064, 4303
`\MT@afteraftergroup` 1772, 1780, 1790
`\MT@auto` 1442, 4252, 4266, 4277, 4283
`\MT@auto@` 1442, 1451, 1495
`\MT@autofalse` 284, 4264, 4276
`\MT@autotrue` 284, 3958
`\MT@babelfalse` 291
`\MT@babeltrue` 291
`\MT@begin@catcodes` 2206, 2207, 2247, 3142, 3158,
 3192, 3208, 3230, 3251, 3265, 3528, 3973, 3974
`\MT@cat` 1347, 2660, 2662

- \MT@cfg@catcodes 857, 1369, 2232, 2249
- \MT@char 1161, 1162, 1172,
1173, 1178, 1179, 1182, 1184, 1199, 1200,
1203, 1376–1380, 1481, 1482, 1484, 1485,
1522–1524, 1553, 1554, 1560, 1561, 1567,
1568, 1571, 1572, 1577–1582, 1632, 1633,
1639, 1640, 1643, 1644, 1649–1653, 2163,
2369, 2385, 2393, 2395, 2405, 2407, 2415,
2416, 2424, 2425, 2430, 2432, 2438, 2440,
2443, 2447, 2451, 2455, 2567, 2570, 2572,
2616, 2619, 2622, 2624, 3578–3580, 3585–3587
- \MT@char@ 1205,
1206, 1209, 1244, 2369, 2375, 2380, 2385,
2433, 2504, 2506, 2512, 2513, 2515, 2528,
2529, 2532, 2533, 2536, 2537, 2541, 2543,
2571, 2590, 2593, 2596, 2605, 2607, 2620, 2634
- \MT@charstring 2382, 2570, 2575
- \MT@charxstring 2600, 2624
- \MT@check@active@set
. 3984, 4202, 4320, 4352, 4372, 4399
- \MT@check@font 915, 2745, 2813
- \MT@check@font@cx 2750, 2813
- \MT@check@range 3707, 3709
- \MT@check@range@ 3709, 3710
- \MT@check@rlist 3660, 3700
- \MT@check@rlist@ 3700, 3701
- \MT@check@step 4308
- \MT@checklist@ 1000, 1015, 2149
- \MT@checklist@family 1033
- \MT@checklist@font 1077
- \MT@checklist@size 1061
- \MT@checksetup 4014, 4025, 4032, 4051, 4081
- \MT@clear@options 161, 274, 726, 3947, 4150
- \MT@clist@break 569, 1028, 1056, 1072, 1086
- \MT@clist@function 569
- \MT@cnt@encoding 3599, 3607, 3608
- \MT@cnt@family 3605, 3614, 3615
- \MT@cnt@series 3612, 3621, 3622
- \MT@cnt@shape 3619, 3627, 3628
- \MT@config@file 3962, 3971, 3972, 3976, 3977, 3980
- \MT@context 2290, 2302, 2321, 2348
- \MT@copy@font 902,
937, 3389, 3395, 3407, 3414, 3865, 3867, 4186
- \MT@copy@font@ 937, 3389, 3395, 3407, 3414, 3865
- \MT@count 486, 487,
675, 1191, 1199, 1201, 1203, 1207, 1209,
1212, 1218, 1223, 1224, 1228, 1239, 1270, 1332
- \MT@curr@file 2210, 2211, 2220, 2221,
3144, 3145, 3363, 3365, 3368, 3550, 3976, 4538
- \MT@curr@list@name 1237, 1273, 1363, 1491,
2160, 2191, 2359, 2629, 2637, 2642, 2650, 2656
- \MT@curr@ls 1751, 1852, 1855
- \MT@curr@set@name
2873, 2875–2878, 2880, 2881, 2886, 2891,
2895, 2896, 2924, 2928, 2957, 2987, 2993,
2999, 3003, 3004, 3245, 3321, 3342, 3357,
3376, 3448, 3452, 3459, 3462, 3464, 3470,
3473, 3476, 3488, 3501, 3509, 3517, 3551, 3553
- \MT@declare@char@inh 3533, 3538, 3548
- \MT@declare@sets 2856, 2862, 2874, 3165
- \MT@DeclareSet 2851, 2853, 2870
- \MT@DeclareSetAndUseIt 2850, 2869
- \MT@DeclareVariants 3110, 3111, 3113
- \MT@def@bool@opt 3820,
3833, 3834, 3837, 3851, 3863, 3873, 3885
- \MT@def@n 369, 2844, 2845
- \MT@default@ex@set 3095
- \MT@default@kn@set 3095
- \MT@default@pr@set 3095
- \MT@default@sp@set 3095
- \MT@default@tr@set 3095
- \MT@define@code@key
. 3289, 3349, 3351, 3352, 3560, 3562, 3563
- \MT@define@code@key@family 3300, 3350, 3561
- \MT@define@code@key@font 3326, 3354, 3565
- \MT@define@code@key@size 3314, 3353, 3564
- \MT@define@opt@key 3355, 3378–3381, 3513–3515
- \MT@define@optionX 4002, 4060, 4063
- \MT@define@optionX@ 4071, 4100, 4102, 4103
- \MT@define@set@key@ 2884, 3043–3046
- \MT@define@set@key@font 2991, 3048
- \MT@define@set@key@size 2917, 3047
- \MT@detokenize@c 410, 2377, 2568
- \MT@detokenize@n 410, 2618
- \MT@dimen@six
. 1121, 1192, 1228, 1270, 1936, 1939, 2114
- \MT@dinfo 87
- \MT@dinfo@list
1013, 1023, 1026, 1031, 1040, 1043, 1049,
1051, 1059, 1067, 1070, 1075, 1085, 1088, 1092
- \MT@dinfo@nl 87
- \MT@DisableLigatures 3155
- \MT@do@font 659, 1138, 1455, 1590, 1660, 1744
- \MT@documentfalse 323
- \MT@documenttrue 323, 4482
- \MT@dofalse 992, 1004, 1027, 1044, 1071, 1089
- \MT@dotrue 992, 995, 1024, 1041, 1050, 1068, 2144
- \MT@draftfalse 287, 3887
- \MT@draftrue 287, 3889
- \MT@edef@n 371, 2326, 2352, 2838,
3295, 3309, 3365, 3368, 3371, 3549, 3687, 3919
- \MT@encoding 970, 1081, 1102, 1104, 1107,
2285, 2297, 2377, 2568, 2617, 2641, 2648, 2656
- \MT@end@catcodes 2208, 2251, 3150,
3167, 3203, 3227, 3246, 3262, 3276, 3543, 3975
- \MT@endinput 165, 3942, 3951
- \MT@engine 173,
255, 259, 265, 271, 2045, 2049, 3943, 3949
- \MT@engine@toold 173, 256
- \MT@error 76, 718,
805, 2045, 2056, 2183, 2196, 2955, 3074,
3101, 3173, 3398, 3424, 3431, 3875, 4037,
4055, 4089, 4257, 4270, 4339, 4410, 4423, 4499
- \MT@error@doesnt@work 4406
- \MT@ex@c@name 1424,
1426, 1497, 1498, 1508, 1514, 1515, 1527, 3211
- \MT@ex@context 941, 2805, 2843
- \MT@ex@doc@contexts 2843
- \MT@ex@factor 292, 1443, 1501
- \MT@ex@factor@ 1443, 1457, 1461, 1471, 1472, 1495
- \MT@ex@inh@name 1483–1485
- \MT@ex@level 292, 4303, 4305
- \MT@ex@max 304, 1474, 1475
- \MT@ex@min 304, 1477, 1478
- \MT@ex@setname 3066
- \MT@ex@split@val 1469
- \MT@exp@cs 366, 369, 372, 375, 378, 952,
954, 963, 1065, 1183, 1257, 1260, 1290,
1294, 1304, 1307, 1310, 1485, 1572, 1644,

- 1792–1794, 2313, 2768, 2776, 2777, 2781,
2824, 2835, 2923, 3320, 3340, 3587, 3662, 3705
- \MT@exp@gcs 366, 370, 374, 376, 379
- \MT@exp@one@n 381, 415, 576,
714, 906, 908, 964, 1019, 1082, 1687, 2215,
2745, 2747, 2753, 2765, 2787, 2821, 2832,
2843, 2862, 3042, 3059, 3088, 3137, 3538, 4487
- \MT@exp@two@c 383, 401, 406,
416, 611, 905, 945, 947, 949, 958, 1829,
1864, 1867, 1870, 2376, 2381, 2569, 2622, 2623
- \MT@exp@two@n 385, 1037, 1046, 2890, 2998
- \MT@expandfont 1428, 1446, 1450
- \MT@expansion 921, 1415, 4330
- \MT@expansionfalse 283, 4213, 4297
- \MT@expansiontrue 283, 3957
- \MT@extra@context 2843, 3195, 3211,
3234, 3254, 3268, 3335, 3338, 3339, 3341,
3382, 3390, 3396, 3408, 3413, 3524, 3659,
3663, 3666, 3669, 3672, 3676, 3677, 3679, 3706
- \MT@extra@inputenc 3522, 3547, 3552, 3553
- \MT@factor@default 316, 3926, 4199
- \MT@family 906, 970,
1038, 1081, 1102, 1104, 1107, 2286, 3129, 3130
- \MT@familyalias
. 907, 908, 981, 1045, 1047, 2296, 2298, 3131
- \MT@feat 992, 1126, 1164, 1252, 1254,
1256, 1257, 1259, 1260, 1266, 1268, 1271,
1279–1283, 1285, 1287–1291, 1294, 1295,
1300, 1304, 1307, 1310, 1313–1315, 1324,
1325, 1330, 1348, 1833, 2181, 2183, 2186,
2187, 2191, 2193, 2196, 2322, 2324, 2326,
2330, 2334, 2338, 2339, 2342, 2350, 2352,
2353, 2355, 2362, 2366, 2630, 2631, 2660, 2662
- \MT@features
711, 2843, 2856, 3042, 3053, 3082, 3360, 3533
- \MT@features@long 711, 714, 719, 2818, 3545
- \MT@file@list 2202,
2204, 2212, 2215, 2217, 2222, 2225, 3137, 3141
- \MT@find@file 906, 908, 2202
- \MT@fix@catcode 5
- \MT@fix@font@set 2909, 4170
- \MT@font 380, 671, 787,
817, 905, 958, 965, 1122, 1129, 1136, 1137,
1172, 1173, 1178, 1179, 1199, 1200, 1203,
1206, 1209, 1234, 1239, 1244, 1338, 1341,
1376, 1377, 1379, 1380, 1451, 1455, 1481,
1482, 1522, 1524, 1553, 1554, 1560, 1561,
1567, 1568, 1577–1579, 1581, 1582, 1587–
1589, 1632, 1633, 1639, 1640, 1649, 1650,
1652, 1653, 1658, 1659, 1687, 1691, 1749,
1750, 2153, 2392, 2446, 2664, 2716–2718,
2720, 2733, 2734, 2745, 2746, 2753, 2779, 2788
- \MT@font@copy 941, 942, 949, 950, 958–960
- \MT@font@list 2664, 2745–2748, 2812
- \MT@font@orig 942
- \MT@font@sets 2894, 2909, 3002, 4170
- \MT@fontspecfalse 783
- \MT@fontspectrue 783, 785, 814
- \MT@gdef@n 369, 3186, 3202,
3226, 3261, 3275, 3464, 3488, 3501, 3555, 3992
- \MT@get@axis 3016–3019, 3029
- \MT@get@basefamily 2214, 2252
- \MT@get@basefamily@ 2258, 2261
- \MT@get@char@unit 1163, 1302, 1328, 1420
- \MT@get@charwd 1197, 1228, 1305, 1328
- \MT@get@config 3962
- \MT@get@ex@opt 1419, 1437, 1495
- \MT@get@ex@opt@ 1503–1505, 1507, 1513
- \MT@get@font 2997, 3007
- \MT@get@font@ 3008, 3013, 3347
- \MT@get@font@and@size 3331, 3346
- \MT@get@font@dimen 1232, 1308
- \MT@get@font@dimen@six 1110, 1121, 1537, 1616, 1703
- \MT@get@highlevel ... 2889, 2899, 3031, 3294, 3305
- \MT@get@inh@list ... 1113, 1422, 1540, 1619, 2348
- \MT@get@listname 2267, 2324, 2350
- \MT@get@listname@ 2267
- \MT@get@ls@basefont 1718, 1854, 1861
- \MT@get@opt 1111, 1277, 1538, 1617
- \MT@get@range 2921, 2931, 3318
- \MT@get@size 2938, 2943, 2952, 2971, 3027
- \MT@get@slot 1160, 2162, 2369, 3577, 3584
- \MT@get@slot@ 2386, 2390
- \MT@get@space@unit
. 1302, 1404, 1551, 1558, 1565, 1630, 1637
- \MT@get@tr@opt 1705, 1799
- \MT@get@tr@opt@ 1817–1820, 1822
- \MT@get@unit 1310, 1318, 1810
- \MT@get@unit@ 1318
- \MT@getkey 4137, 4152
- \MT@glet 364, 376, 945, 1793, 1864,
1895, 2050, 2058, 2714, 2813–2816, 3131,
3389, 3407, 3602, 3865, 3867, 4451, 4456, 4542
- \MT@glet@nc 375, 1107, 1908, 2105, 2362,
2734, 2812, 2878, 2880, 2886, 2993, 3219,
3448, 3452, 3459, 3462, 3470, 3473, 3476, 3509
- \MT@glet@nn 378, 2831, 3989
- \MT@if@false 281,
788, 795, 819, 836, 2758, 3704, 4518, 4525
- \MT@if@list@exists
. 1100, 1418, 1436, 1536, 1615, 1704, 2321
- \MT@if@outer@next
. 1985, 2000, 2004, 2007, 2012, 2017–2020
- \MT@if@true 281,
789–791, 796–798, 820–822, 849, 853, 854,
2751, 3713, 3717, 3725, 3730, 4519–4523, 4526
- \MT@ifdefined@c@T 389,
792, 825, 1181, 1217, 1483, 1570, 1642,
1737, 2556, 3129, 3552, 3649, 3998, 4001, 4545
- \MT@ifdefined@c@TF
. 389, 748, 1708, 1721, 1724, 1735,
1852, 1933, 2111, 2157, 2158, 2325, 2351, 3658
- \MT@ifdefined@n@T
. 389, 737, 1016, 1034, 1062, 1078, 1182,
1313, 1349, 1484, 1508, 1571, 1643, 1802,
1804, 1823, 2304, 2659, 2876, 3124, 3214,
3372, 3608, 3615, 3622, 3628, 3659, 3669, 4137
- \MT@ifdefined@n@TF
. 389, 981, 998, 1017, 1035, 1063,
1079, 1102, 1279, 1287, 1497, 1514, 1791,
2147, 2186, 2292, 2377, 3068, 3071, 3097,
3363, 3635, 3686, 3695, 3761, 3985, 3988, 4485
- \MT@ifdim .. 509, 631, 632, 636, 637, 2954, 2963,
3711–3713, 3715, 3716, 3723–3725, 3728, 3729
- \MT@ifdimen 471, 2982, 3451, 3461, 3475, 3935
- \MT@ifempty 422, 1136, 1137, 1170, 1176,
1393, 1394, 1550, 1557, 1564, 1587–1589,
1601–1603, 1605–1607, 1629, 1636, 1658,
1659, 1675, 1676, 1914, 1915, 1925, 1943,
2080, 2103, 2104, 2116, 2823, 2855, 2860,

- 2901, 2907, 2932, 2933, 2946, 2947, 3021,
 3032, 3052, 3057, 3081, 3086, 3162, 3241,
 3356, 3362, 3382, 3388, 3396, 3406, 3413,
 3486, 3532, 3537, 3774, 3805, 3965, 4011, 4078
 \MT@ifint 435, 2535, 3242, 3487, 3918, 3928
 \MT@ifstreq . 518, 1355, 2182, 2323, 2338, 2591,
 2594, 2606, 2786, 2996, 3330, 3450, 3472,
 3474, 3650, 3777, 3779, 3782, 3785, 3807,
 3809, 3823, 3824, 3895, 3896, 3901, 3905,
 3934, 4013, 4020, 4024, 4031, 4080, 4086, 4382
 \MT@in@clist 597,
 714, 738, 1019, 1037, 1046, 1082, 1687,
 2204, 2215, 2745, 2747, 2753, 2765, 2821, 3137
 \MT@in@rlist 625, 1065, 2313
 \MT@in@rlist@ 625
 \MT@in@rlist@@ 625
 \MT@in@tlist 613, 2264, 2832
 \MT@in@tlist@ 613
 \MT@inannotfalse 100
 \MT@inannottrue 100
 \MT@increment 675, 3607, 3614, 3621, 3627
 \MT@info 76, 91, 96, 3902, 4121, 4125
 \MT@info@missing@char 1212, 1242, 2442
 \MT@info@nl 76, 92, 93, 97, 749, 1103,
 1243, 3893, 3903, 3972, 3986, 3990, 4182,
 4186, 4198, 4205, 4304, 4331, 4351, 4356,
 4365, 4375, 4398, 4402, 4415, 4430, 4495, 4506
 \MT@inh@do 2361, 3566
 \MT@inh@feat 3522, 3532, 3535, 3546
 \MT@inh@split 3568, 3572
 \MT@inlist@false 597, 601, 614, 626, 2761
 \MT@inlist@true 597, 603, 621, 633, 638, 2761
 \MT@is@active 2376, 2548
 \MT@is@char 2381, 2569, 2575
 \MT@is@charx 2584, 2600, 2623
 \MT@is@composite 2379, 2614
 \MT@is@feature 713, 2861, 3058, 3087
 \MT@is@letter 2374, 2492, 2572, 2619
 \MT@is@number 2518, 2523
 \MT@is@symbol 2378, 2566
 \MT@iterate 647
 \MT@kerning 925, 1612, 4401
 \MT@kerningfalse 289
 \MT@kerningtrue 289
 \MT@kn@cc@name 1621, 1623, 1666, 3268
 \MT@kn@context 2806, 2843
 \MT@kn@doc@contexts 2843
 \MT@kn@factor 292
 \MT@kn@factor@ 1279
 \MT@kn@inh@name 1642–1644
 \MT@kn@max 304
 \MT@kn@min 304
 \MT@kn@setname 3066
 \MT@kn@split@val 1627
 \MT@kn@unit 298
 \MT@kn@unit@ 1287, 1669
 \MT@led@unh@box@line 749
 \MT@led@mac@setup 743, 811, 812
 \MT@let@cn 377, 378, 379, 982, 1117, 1426,
 1498, 1544, 1623, 1803, 1805, 1809, 2181,
 2193, 2318, 2322, 2355, 2716, 2975, 3376, 3551
 \MT@let@nc . . 375, 1315, 2026, 2334, 2366, 2756,
 3688, 3693, 3783, 3786, 4027, 4034, 4413, 4428
 \MT@let@nn
 378, 1280, 1285, 1288, 1300, 1515, 1518, 1824
 \MT@letterspace
 . . . 319, 1708, 1803, 1813, 4459, 4460, 4462
 \MT@letterspace@
 1147, 1217, 1218, 1708, 1709, 1712,
 1715, 1719, 1723, 1731, 1743–1745, 1751,
 1760, 1939, 1943, 2066, 2081–2084, 2117, 2120
 \MT@letterspace@default 319, 4460
 \MT@listname
 2267, 2315, 2318, 2325, 2326, 2328, 2330,
 2351, 2352, 2354, 2355, 2359, 2362, 3587, 3592
 \MT@listname@count 3359, 3364, 3366
 \MT@load@inputenc 1356, 1359, 1368
 \MT@load@list 1115, 1424, 1542, 1621, 2178
 \MT@loop 647, 655, 669, 3692
 \MT@lower 2925, 2931, 3321
 \MT@ls@adjust 1758, 2077
 \MT@ls@adjust@ 2061, 2085
 \MT@ls@adjust@empty 2063, 2077
 \MT@ls@adjust@relax 2064, 2077
 \MT@ls@basefont 1861, 1870, 1873, 1874
 \MT@ls@outer@k 1761,
 1766, 1774, 1956, 1970, 2023, 2040, 2125
 \MT@ls@set@ls 2068, 2071, 2077
 \MT@ls@too@large 2084, 2087, 4462
 \MT@ls@font 1714, 1719,
 1731, 1732, 1741, 1744, 1745, 1749, 1829,
 1839, 1884, 1886, 1901, 1902, 1909, 1910,
 1916, 1920, 1927, 1937, 1941, 1944, 2118, 2121
 \MT@lslig 1851
 \MT@lua 242, 459, 496, 541, 2397
 \MT@luatex@no 240, 332
 \MT@map@clist@ 569
 \MT@map@clist@c 569, 951, 2161, 2752,
 2764, 2775, 2811, 2818, 2856, 2858, 3053,
 3055, 3082, 3084, 3360, 3533, 3535, 3545, 4135
 \MT@map@clist@n 569, 997, 2146,
 2843, 2887, 2919, 2994, 3042, 3114, 3184,
 3292, 3303, 3316, 3328, 3468, 3484, 3581,
 3769, 3772, 3801, 3803, 3832, 3913, 4009, 4076
 \MT@map@tlist@ 587
 \MT@map@tlist@c 587, 616, 627, 954, 1183,
 1485, 1572, 1644, 2271, 2781, 3660, 3705, 4170
 \MT@map@tlist@n 587, 2243, 2577, 3601
 \MT@max@char 2492, 2515
 \MT@max@slot 663, 2492, 2541
 \MT@maybe@do . . . 992, 1097, 1416, 1534, 1613, 1689
 \MT@maybe@etex 149
 \MT@maybe@rem@from@list 2781, 2785
 \MT@MT 2, 76, 79, 82, 84,
 86, 91, 92, 94, 95, 270, 725, 806, 808, 3398,
 3945, 3966, 4121, 4125, 4183, 4386, 4499, 4500
 \MT@next@listname 2274, 2283
 \MT@next@listname@ 2293, 2305, 2311
 \MT@nl@ligatures 2153, 3155
 \MT@nl@setname 2145, 3155
 \MT@nofamilyfalse 1094, 1099
 \MT@nofamilytrue 1094, 2284
 \MT@noligatures 929, 2141, 4471
 \MT@noligatures@ 1884, 1886, 2153, 2156
 \MT@noligaturesfalse 286
 \MT@noligaturestrue 286, 3164
 \MT@nonselectedfalse 1432, 1447
 \MT@nonselectedtrue 1432, 1435
 \MT@noresetfalse
 . . 2501, 2508, 2517, 2538, 2591, 2594, 2606

- `\MT@noresttrue` 2372, 2501, 2526
`\MT@old@cmd` 66, 71–74
`\MT@opt@autofalse` 3744
`\MT@opt@autotruer` 3744, 3834
`\MT@opt@def@set` 3760, 3788, 3813
`\MT@opt@DVI@false` 3745
`\MT@opt@DVI@true` 3745, 3840, 3843
`\MT@opt@expansion@false` 3743
`\MT@opt@expansion@true` 3743
`\MT@options` 3279
`\MT@optwarn@admissible` 3746, 3825, 3905
`\MT@optwarn@nan` 3750, 3920, 3930
`\MT@orig@add@accent` 2735
`\MT@orig@foreign@language` 4512, 4514
`\MT@orig@pickupfont` 842, 885, 890, 2670, 2737
`\MT@orig@py@macron` 882, 886, 891
`\MT@orig@select@language` 4507, 4509
`\MT@outer@kern` 1760, 1764, 1766, 1767, 1771, 1779,
1855, 1856, 1858, 2023, 2024, 2040, 2098, 2137
`\MT@outer@space`
1754, 1755, 1899, 1952, 1954, 1955, 1968,
1969, 1992, 1993, 2008, 2009, 2013, 2014, 2038
`\MT@pdf@annot` 99
`\MT@pdf@or@lua` 178, 183, 268, 438
`\MT@pdf@tex@no` 199, 331
`\MT@permute` 3201, 3225, 3239, 3260, 3274, 3558, 3597
`\MT@permute@` 3597
`\MT@permute@@` 3597
`\MT@permute@@@` 3597
`\MT@permute@@@@` 3597
`\MT@permute@@@@@` 3626, 3631
`\MT@permute@@@@@@` 3639, 3641
`\MT@permute@define` 3632, 3642–3644, 3684
`\MT@permute@reset` 3601, 3690
`\MT@permutelist`
. 3195, 3211, 3237, 3254, 3268, 3335,
3336, 3341, 3556, 3654, 3655, 3659, 3663,
3666, 3669–3672, 3679, 3680, 3706, 3737, 3738
`\MT@pickupfont` 887, 892, 2731, 2739
`\MT@plain` 125, 146
`\MT@pr@c@name` 1115, 1117, 1384, 3195
`\MT@pr@c@context` 941, 2805, 2843
`\MT@pr@d@doc@contexts` 2843
`\MT@pr@f@factor` 292, 3929, 4199, 4200
`\MT@pr@f@factor@` 1279
`\MT@pr@inh@name` 1181, 1182, 1184
`\MT@pr@l@level` 292, 4197, 4198
`\MT@pr@l@max` 304
`\MT@pr@l@min` 304
`\MT@pr@s@set@name` 3066
`\MT@pr@s@split@val` 1168
`\MT@pr@u@unit` 298, 3936, 4201
`\MT@pr@u@unit@` 1287, 1387
`\MT@pr@s@reset@aux` 1389,
1391, 1393, 1394, 1397, 1671, 1673, 1675, 1676
`\MT@pr@s@reset@aux@factor` 1389, 1397, 1601–1603, 1671
`\MT@pr@s@reset@aux@space` . 1391, 1397, 1605–1607, 1673
`\MT@pr@s@reset@ex` 1509, 1526
`\MT@pr@s@reset@kn` 1664
`\MT@pr@s@reset@kn@e` 1664
`\MT@pr@s@reset@pr` 1382
`\MT@pr@s@reset@pr@e` 1382
`\MT@pr@s@reset@sp` 1594
`\MT@pr@s@reset@sp@e` 1594
`\MT@ProcessOptionsWithKV` 4129, 4153
`\MT@protrudechars` 4060, 4064, 4197, 4353
`\MT@protrusion` 920, 1096, 4204
`\MT@protrusion@false` 282, 1787
`\MT@protrusion@true` 282, 3952
`\MT@rbba@expansion` 706
`\MT@rbba@kerning` 706
`\MT@rbba@protrusion` 706
`\MT@rbba@spacing` 706
`\MT@rbba@tracking` 706
`\MT@register@font` 933, 2746, 2814
`\MT@register@font@cx` 2774, 2814
`\MT@register@subst@font` 2722, 2747, 2815
`\MT@register@subst@font@cx` 2763, 2815
`\MT@rem@from@clist` 608, 964, 2787, 3284
`\MT@rem@from@list` 954, 962
`\MT@rem@last@space` 410
`\MT@repeat` 647, 657, 671, 3698
`\MT@requires@latex` 145, 162, 729, 838,
869, 1828, 2668, 3752, 3912, 4130, 4134, 4541
`\MT@requires@luatex` 326, 912, 1145, 1683, 4073, 4347
`\MT@requires@pdftex` 326,
437, 472, 519, 744, 901, 911, 924, 928,
939, 1144, 1189, 1215, 1459, 1533, 1612,
1682, 1881, 2142, 3156, 3386, 3500, 3508,
3862, 3956, 4072, 4224, 4254, 4282, 4346, 4469
`\MT@res@a` 531, 533, 599, 606, 609, 611, 615, 620
`\MT@res@b` 532, 533, 609–611, 619, 620
`\MT@reset@context` 2795, 2799, 2803, 2830
`\MT@reset@context@` 2803, 2830
`\MT@reset@ef@codes` 1421, 1445, 1458, 1510
`\MT@reset@ef@codes@` 1453, 1462, 1467
`\MT@reset@kn@codes` 1618, 1625, 1655
`\MT@reset@kn@codes@` 1655
`\MT@reset@pr@codes` 1112, 1119, 1140
`\MT@reset@pr@codes@` 1140
`\MT@reset@sp@codes` 1539, 1546, 1584
`\MT@reset@sp@codes@` 1584
`\MT@restore@catcodes` 5, 7, 8, 163, 4547
`\MT@restore@p@h` 779, 799, 855
`\MT@s@saved@setupfont` 4118, 4126
`\MT@s@scale` 682, 1192, 1253, 1267, 1270, 1472
`\MT@s@scale@factor` 1194, 1225, 1251, 1399, 1528
`\MT@s@scale@to@em` . . 1171, 1177, 1189, 1222, 1405,
1552, 1559, 1566, 1631, 1638, 1812, 1934, 2112
`\MT@s@scrubfeature` 974, 985, 3307
`\MT@s@scrubfeatures` 985, 2913, 3333
`\MT@s@selected@false` 285
`\MT@s@selected@true` 285
`\MT@s@series` 970, 1081, 2287, 2299
`\MT@s@set@all@ex` 1453, 1529
`\MT@s@set@all@kn` 1655, 1677
`\MT@s@set@all@pr` 1133, 1140, 1395
`\MT@s@set@all@sp` 1584, 1609
`\MT@s@set@babel@context` 4484, 4510, 4515, 4529
`\MT@s@set@codes` . 1118, 1151, 1427, 1545, 1624, 2194
`\MT@s@set@curr@ls` 1751
`\MT@s@set@curr@ook` 1771, 1773, 1779, 1782, 1997, 2001
`\MT@s@set@curr@os` 1753, 1781, 1997, 2001
`\MT@s@set@default@set` 3082, 3088, 3095
`\MT@s@set@ex@codes` 1449, 4288, 4290
`\MT@s@set@ex@codes@n` 1432, 1449, 4290
`\MT@s@set@ex@codes@s` 1417, 4288
`\MT@s@set@ex@heirs` 1485, 1521
`\MT@s@set@inh@list` 3529, 3531
`\MT@s@set@inputenc` 1114, 1346, 1423, 1541, 1620, 2360

- \MT@set@inputenc@ 1349, 1351
 \MT@set@kn@codes 1614
 \MT@set@kn@heirs 1644, 1648
 \MT@set@listname 1116,
 1278, 1425, 1496, 1543, 1622, 1801, 2629
 \MT@set@lsbasefont 1870
 \MT@set@lsfont 1748, 1827, 1839
 \MT@set@named@keys 3197, 3213, 3235, 3256, 3270, 3279
 \MT@set@pr@codes 1098
 \MT@set@pr@heirs 1185, 1374
 \MT@set@sp@codes 1535
 \MT@set@sp@heirs 1572, 1576
 \MT@set@tr@codes 1699, 1834, 1844
 \MT@set@tr@zero 1710, 1870, 4100
 \MT@SetExpansion 3209, 3211
 \MT@SetExtraKerning 3266, 3268
 \MT@SetExtraSpacing 3252, 3254
 \MT@SetProtrusion 3193, 3195
 \MT@SetTracking 3231, 3233
 \MT@setup@ 727, 731, 3854, 3855, 4542
 \MT@setup@contexts 2794, 2810
 \MT@setup@copies 4169, 4185
 \MT@setup@expansion 4172, 4209
 \MT@setup@kerning 4176, 4393, 4436
 \MT@setup@noligatures 4177, 4467
 \MT@setup@PDF 4168, 4180, 4190
 \MT@setup@protrusion 4171, 4193
 \MT@setup@spacing 4175, 4360, 4437
 \MT@setup@spacing@check 4378, 4545, 4546
 \MT@setup@tracking 4173, 4345, 4408, 4435
 \MT@setup@warntracking 4174, 4440
 \MT@setupfont .. 899, 2725, 4119, 4122, 4126, 4162
 \MT@setupfont@hook
 786, 815, 817, 824, 829, 855, 862, 900
 \MT@shape 970, 1081, 2288, 2300
 \MT@shorthandoff 4493, 4524, 4527
 \MT@show@pdfannot 106
 \MT@shrink 301, 1440,
 4220, 4221, 4228, 4229, 4232, 4236, 4293, 4306
 \MT@shrink@ 1440, 1451, 1495
 \MT@shrink@default 317
 \MT@size 632, 636, 637, 970, 1081
 \MT@size@name 625, 2315
 \MT@sp@cc@name 1542, 1544, 1596, 3254
 \MT@sp@context 2806, 2843, 4382
 \MT@sp@doc@contexts 2843
 \MT@sp@factor 292
 \MT@sp@factor@ 1279
 \MT@sp@inh@name 1570–1572
 \MT@sp@max 304
 \MT@sp@min 304
 \MT@sp@setname 3066
 \MT@sp@split@val 1548
 \MT@sp@unit 298
 \MT@sp@unit@ 1287, 1599
 \MT@spacing 925, 1532, 4374
 \MT@spacingfalse 288
 \MT@spacingtrue 288
 \MT@split@codes 1153, 1157
 \MT@split@name 905, 970
 \MT@step 301, 1441, 4223,
 4225, 4243, 4247, 4250, 4307, 4310, 4311, 4315
 \MT@step@ 1441, 1451, 1495
 \MT@stretch 301, 1439, 4217,
 4218, 4221, 4228, 4230, 4235, 4238, 4292, 4306
 \MT@stretch@ 1439, 1451, 1495
 \MT@stretch@default 317, 4218
 \MT@strip@prefix 2584, 2600, 2622
 \MT@temp 1135–1138, 1586–1590,
 1657–1660, 1904, 1908, 1911, 1916, 1919,
 1924, 1989, 1992, 1998, 2002, 2005, 2008,
 2013, 2021, 2023, 2029, 2102, 2105, 2122,
 3015, 3040, 4133, 4138, 4143, 4144, 4146, 4149
 \MT@tempencoding 3634, 3635, 3645, 3651
 \MT@tempfamily 3646
 \MT@tempseries 3647
 \MT@tempshape 3648
 \MT@tempsize 3602, 3649, 3658, 3660, 3664
 \MT@test@east 1932, 2110, 2900, 2905
 \MT@textls 49, 2063, 2064, 2066
 \MT@the@pr@code 1142, 1172, 1178, 1746
 \MT@the@pr@code@tr 1142, 1746
 \MT@tlist@break 587, 622, 644, 2276, 3740
 \MT@toks 280, 1243, 1273, 1491, 2373, 2408, 2434,
 2506, 2513, 2560, 2596, 2607, 2634, 2646, 2654
 \MT@tr@cc@name
 ... 1802–1805, 1809, 1823, 1824, 3236, 3243
 \MT@tr@context 2806, 2843
 \MT@tr@doc@contexts 2843
 \MT@tr@factor@ 1811
 \MT@tr@font@list 1685
 \MT@tr@ispace 1721, 1722, 1817
 \MT@tr@ligatures 1737, 1820, 1883, 1886
 \MT@tr@max 304, 2088–2090
 \MT@tr@min 304, 2092–2094
 \MT@tr@noligatures 1737, 1880
 \MT@tr@okern 1735, 1819
 \MT@tr@ospace 1724, 1725, 1817
 \MT@tr@outer@icr 2002, 2031
 \MT@tr@outer@icr@ 2031
 \MT@tr@outer@l 1756, 1948
 \MT@tr@outer@next 1982, 1986, 2033
 \MT@tr@outer@r 1783, 1976
 \MT@tr@outer@r@ 1980, 1981, 1982, 1998, 2005, 2031
 \MT@tr@outer@r@@ 1983, 1988
 \MT@tr@set@okern 1736, 2098
 \MT@tr@set@okern@ 2103, 2104, 2109
 \MT@tr@set@space 1727, 1900
 \MT@tr@set@space@ 1905–1907, 1913
 \MT@tr@set@space@@ 1918, 1923, 1926, 1931
 \MT@tr@setname 3066
 \MT@tr@unit@ 1804, 1933, 2111
 \MT@tracking 913, 1685, 1834, 2727, 4100, 4101, 4355
 \MT@tracking@ 1685, 4101
 \MT@trackingfalse 290, 4412
 \MT@trackingtrue 290
 \MT@try@order 2271, 2279
 \MT@undefined@char 2553, 2555, 2565
 \MT@upper 2925, 2931, 3321
 \MT@use@set 3053, 3059, 3066
 \MT@val 2823–2825, 2827, 2833,
 2835, 2838, 2888, 2892, 2900, 2902, 2920–
 2922, 2934, 2937, 2939, 2942, 2944, 2945,
 2951, 2953, 2954, 2956, 2959, 2961, 2972,
 2975, 2979, 2982–2984, 2986, 2988, 2995–
 2997, 3000, 3009–3011, 3023, 3025, 3030,
 3032, 3034, 3036, 3293, 3295, 3304, 3307,
 3309, 3317–3319, 3329–3331, 3337, 3342,
 3580, 3587, 3591, 3592, 3761, 3762, 3765,
 3773, 3774, 3777, 3779, 3782, 3785, 3804,

- 3805, 3807, 3809, 4010, 4011, 4013, 4020,
4024, 4031, 4037, 4077, 4078, 4080, 4086, 4089
- `\MT@variants` 2264, 3107, 3117
- `\MT@vinfo` 76, 93, 919, 1010, 1282,
1291, 1295, 1330, 1499, 1516, 1702, 1712,
2164, 2173, 2187, 2211, 2221, 2224, 2328,
2330, 2336, 2804, 2827, 3139, 3145, 3893,
3900, 4016, 4022, 4028, 4035, 4083, 4088, 4486
- `\MT@warn@ascii` 2515, 2633
- `\MT@warn@axis@empty` 3022, 3033, 3038
- `\MT@warn@code@too@large` 1257, 1260, 1264
- `\MT@warn@err` 76, 3897, 3898
- `\MT@warn@ex@too@large` 1475, 1478, 1489
- `\MT@warn@maybe@inputenc` ... 2159, 2647, 2655, 2658
- `\MT@warn@nodim` 3440, 3453, 3463, 3477
- `\MT@warn@number@too@large` 2542, 2639
- `\MT@warn@preset@tewidth` ... 1388, 1408, 1600, 1670
- `\MT@warn@rest` 2414, 2450, 2644
- `\MT@warn@tracking@DVI` 1713, 4442
- `\MT@warn@unknown` ... 2404, 2420, 2429, 2454, 2652
- `\MT@warning` 67, 76, 94, 2089, 2093,
2167, 2339, 2877, 2986, 3039, 3125, 3147,
- 3244, 3373, 3442, 3490, 3504, 3652, 3670,
3737, 3897, 3902, 3998, 4069, 4108, 4110, 4111
- `\MT@warning@nl` 76, 88, 95, 257,
724, 759, 770, 831, 1123, 1235, 1271, 1323,
1363, 1409, 1490, 1892, 2634, 2640, 2645,
2653, 2698, 3216, 3415, 3747, 3754, 3765,
3846, 3898, 3903, 3937, 3944, 3979, 3993,
4159, 4248, 4294, 4313, 4367, 4383, 4446, 4531
- `\MT@while@num` 653, 661, 858–860
- `\MT@with@babel@and@T`
... 736, 789–791, 820–822, 4519–4523, 4526
- `\MT@with@package@T`
733, 784, 785, 793, 796–798, 804, 811–814,
827, 839, 853, 854, 856, 870, 881, 2709, 4366
- `\MT@xadd` 104, 553, 2212, 2217, 2222,
2225, 2768, 2777, 2835, 2923, 3141, 3320, 3587
- `\MT@xaddb` 561, 3340, 3662
- `\MT@xdef@n` .. 371, 3069, 3072, 3099, 3104, 3128,
3243, 3335, 3357, 3517, 3553, 3679, 3762, 3764
- `\MT@xetex@no` 231
- `\MT@xspace` 2021, 2036
- `\MT@xunicodfalse` 782
- `\MT@xunicodtrue` 782, 784, 813
- `\nullfont` 898
- `\pdftracingfonts` 335
- `\pickup@font` 842, 885, 887,
890, 892, 2697, 2699, 2708, 2731, 2737, 2739
- `pifont` (package) 94
- `pinyin` (package) 53, 227, 228
- `protrusion` (option) 6, 11, 118
- `PSNFSS` (package) 188
- `pstricks` (package) 9, 126
- `pxfonts` (package) 21, 135
- N** `\normalfont` 898, 4323, 4327
- P** `\pdfadjustinterwordglue` 4102, 4364
- `\pdfadjustspacing` 4065, 4326
- `\pdfappendkern` 4104, 4397
- `\pdfcopyfont` 949
- `pdfcpot` (package) 26, 29, 52, 61, 93, 222
- `\pdffontexpand` 1451
- `\pdfnoligatures` 2166, 2172
- `\pdfprependkern` 4103, 4396
- `\pdfprotrudechars` 4064, 4455
- `pdftexcmds` (package) 36, 228
- Q** `qfonts` (package) 21, 135, 224
- R** `ragged2e` (package) 26, 77, 130, 228, 229
- `reysize` (package) 101
- S** `selected` (option) 8, 14, 32, 66, 85, 118, 223
- `\selectfont` 1835, 2798, 2808, 4003, 4537
- `\set@fontsize` 2979
- `\SetExpansion` 14, 36, 3206
- `\SetExtraKerning` 18, 38, 3264
- `\SetExtraSpacing` 19, 39, 3249
- `\SetProtrusion` 12, 35, 3190
- `\SetTracking` 15, 37, 3229
- `\sfcode` 4381
- `\shbscode` 1567, 1568, 1579, 1582, 1589
- `\showhyphens` 4321, 4325
- `\rightmargin` 755
- `\rPCODE` 1137, 1178, 1179, 1377, 1380, 1745
- `shrink` (option) .. 8, 15, 32, 120, 127, 129, 225, 227
- `slantsc` (package) 167
- `soul` (package) 4, 29, 53, 78, 226, 228
- `soulutf8` (package) 29, 228
- `\spacefactor` 1959
- `\spaceskip` 1960, 1963
- `spacing` (option) 6, 11, 26, 30, 118, 228, 229
- `\stbscode` 1560, 1561, 1578, 1582, 1588
- `step` (option) . 8, 30, 32, 120, 127, 129, 223, 225, 227
- `stretch` (option) ... 8, 15, 25, 32, 120, 127, 129, 227
- `trace` (package) 95, 226
- `\tracingmicrotype` 87
- `\tracingmicrotypeinpdf` 98
- `\tracingmicrotypeinpdfallfalse` 105
- `\tracingmicrotypeinpdfalltrue` 105
- `tracking` (option) 6, 11, 23, 30, 118
- `txfonts` (package) 21, 135
- T** `\tagcode` 1741, 2158, 2163
- `tex4ht` (package) 51, 226
- `\textls` 23,
52, 53, 847, 872, 877, 1759, 1831, 1847, 2061
- `\textls*` 23
- `\textmicrotypecontext` 22, 48, 843, 2802
- `tikz` (package) 77, 78, 217, 228
- `tipa` (package) 94
- U** `ucs` (package) 90, 227
- `ulgothic` (package) 21
- `unit` (option) 7, 31, 224, 226
- `\UseMicrotypeSet` 11, 33, 60, 73, 2871, 3050

V	verbose (option)	9, 30, 31, 120, 223, 224, 227
W	wordcount (package)	49, 227
X	\XeTeXcharglyph	2440
	\XeTeXcountglyphs	671
	\XeTeXfonttype	2392
	\XeTeXglyph	1206
	\XeTeXglyphindex	2427
	\XeTeXglyphname	2446
Y	yfonts (package)	24
Z	zefonts (package)	21, 135
	\XeTeXprotrudechars	4068
	\xspace	2020, 2021
	xspace (package)	79, 229
	\xspaceskip	1962, 1965
	xunicode (package)	91, 92, 229

E The L^AT_EX Project Public License

LPPL Version 1.3c 2006-05-20

Copyright 1999, 2002–2006 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 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

1. Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
2. You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.
3. You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
4. If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Maintainer are considered to be updated versions of the Work.
5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
 - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
 - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
 - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
- (d) You distribute at least one of the following with the Derived Work:
 - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
 - ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.
10. (a) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.
 - (b) If a Derived Work is distributed under a different license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the Work.
11. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.
12. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless required by applicable law or agreed to in writing will The Copyright Holder, or any au-

thor named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

Maintenance of The Work

The Work has the status ‘author-maintained’ if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that it is ‘author-maintained’.

The Work has the status ‘maintained’ if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status ‘maintained’ to ‘unmaintained’ if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.
2. If this search is successful, then enquire whether the Work is still maintained.
 - (a) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.
 - (b) If the search is unsuccessful or no action to resume active maintenance is taken by the Current

Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a L^AT_EX work, this could be done, for example, by posting to `comp.text.tex`.)

3. (a) If the Current Maintainer is reachable and agrees to pass maintenance of the Work to you, then this takes effect immediately upon announcement.
 - (b) If the Current Maintainer is not reachable and the Copyright Holder agrees that maintenance of the Work be passed to you, then this takes effect immediately upon announcement.
4. If you make an ‘intention announcement’ as described in 2b above and after three months your intention is challenged neither by the Current Maintainer nor by the Copyright Holder nor by other people, then you may arrange for the Work to be changed so as to name you as the (new) Current Maintainer.
5. If the previously unreachable Current Maintainer becomes reachable once more within three months of a change completed under the terms of 3b or 4, then that Current Maintainer must become or remain the Current Maintainer upon request provided they then update their communication data within one month.

A change in the Current Maintainer does not, of itself, alter the fact that the Work is distributed under the LPPL license.

If you become the Current Maintainer of the Work, you should immediately provide, within the Work, a prominent and unambiguous statement of your status as Current Maintainer. You should also announce your new status to the same pertinent community as in 2b above.

Whether and How to Distribute Works under This License

This section contains important instructions, examples, and recommendations for authors who are considering distributing their works under this license. These authors are addressed as ‘you’ in this section.

Choosing This License or Another License

If for any part of your work you want or need to use *distribution* conditions that differ significantly from those in this license, then do not refer to this license anywhere

in your work but, instead, distribute your work under a different license. You may use the text of this license as a model for your own license, but your license should not refer to the LPPL or otherwise give the impression that your work is distributed under the LPPL.

The document ‘`modguide.tex`’ in the base L^AT_EX distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing L^AT_EX under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated

to L^AT_EX, the discussion in ‘modgui.de.tex’ may still be relevant, and authors intending to distribute their works under any license are encouraged to read it.

A Recommendation on Modification Without Distribution

It is wise never to modify a component of the Work, even for your own personal use, without also meeting the above conditions for distributing the modified component. While you might intend that such modifications will never be distributed, often this will happen by accident – you may forget that you have modified that component; or it may not occur to you when allowing others to access the modified version that you are thus distributing it and violating the conditions of this license in ways that could have legal implications and, worse, cause problems for the community. It is therefore usually in your best interest to keep your copy of the Work identical with the public one. Many works provide ways to control the behavior of that work without altering any of its licensed components.

How to Use This License

To use this license, place in each of the components of your work both an explicit copyright notice including your name and the year the work was authored and/or last substantially modified. Include also a statement that the distribution and/or modification of that component is constrained by the conditions in this license.

Here is an example of such a notice and statement:

```
%% pig.dtx
%% Copyright 2005 M. Y. Name
%
% This work may be distributed and/or modified under the
% conditions of the LaTeX Project Public License, either version 1.3
% of this license or (at your option) any later version.
% The latest version of this license is in
% http://www.latex-project.org/lppl.txt
% and version 1.3 or later is part of all distributions of LaTeX
% version 2005/12/01 or later.
%
% This work has the LPPL maintenance status `maintained'.
%
% The Current Maintainer of this work is M. Y. Name.
%
% This work consists of the files pig.dtx and pig.ins
```

```
% and the derived file pig.sty.
```

Given such a notice and statement in a file, the conditions given in this license document would apply, with the ‘Work’ referring to the three files ‘pig.dtx’, ‘pig.ins’, and ‘pig.sty’ (the last being generated from ‘pig.dtx’ using ‘pig.ins’), the ‘Base Interpreter’ referring to any ‘L^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.