

The microtype package

Subliminal refinements towards typographical perfection

R Schlicht
w.m.1@gmx.net

v2.5
2013/03/13

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

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

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

This package is copyright © 2004–2013 R Schlicht. It may be distributed and/or modified under the conditions of the [\$\text{\LaTeX}\$ Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPPL maintenance status ‘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 kerning [70] Tracking [71] Disabling ligatures [80] Loading the configuration [81] Translating characters into slots [85] Hook into L ^A T _E X's font selection [92] Context-sensitive setup [95]	
14.3	Configuration	97
	Font sets [97] Variants and aliases [103] Disabling ligatures [104] Interaction with babel [104] Fine tuning [104] Character inheritance [111] Permutation [113]	

14.4	Package options	116
	Declaring the options [116] Loading the definition file [120] Reading the configuration file [120] Hook for other packages [121] Changing options later [122] Processing the options [124]	
15	Configuration files	133
15.1	Font sets	133
15.2	Font variants and aliases	134
15.3	Interaction with babel	135
15.4	Note on admissible characters	136
15.5	Character inheritance	136
	OT1 [136] T1 [137] LY1 [137] OT4 [138] QX [138] T5 [139] EU1, EU2 [140] Euro symbols [140]	
15.6	Tracking	141
15.7	Font expansion	141
15.8	Character protrusion	143
	Normal [144] Italics [152] Small caps [163] Italic small caps [166] Text companion [167] Computer Modern math [172] AMS symbols [176] Euler [179] Euro symbols [183]	
15.9	Interword spacing	184
	Nonfrenchspacing [186]	
15.10	Additional kerning	187
	French [188] Turkish [188]	
16	OpenType font configuration files	189
17	Auxiliary file for micro fine tuning	189
A	The title logo	191
A.1	Macros	191
A.2	Document	196
B	The letterspacing illustration	196
B.1	Macros	196
B.2	Document	199
C	Change history	201
D	Index	208
E	The L^AT_EX Project Public License	217

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	83

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

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

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

2 Getting started

There is nothing surprising in loading this package:

```
\usepackage{microtype}
```

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

- Enable the respective micro-typographic feature, either via the respective package option or with the `\microtypesetup` command (section 3).
- Select the fonts to which this feature should be applied by declaring and activating ‘sets of fonts’. 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, \langle font set name \rangle *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 \pdfTeX versions older than 1.20, in DVI output mode (see section 3.5), or with \XeTeX . 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 \pdfTeX):

```
\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, \langle font set name \rangle 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 \XeTeX (you may use the ‘LetterSpace’ option of the `fontspec` package instead). With \LuaTeX , you need to load the fonts with the `fontspec` option ‘`Renderer=Basic`’. See the [fontspec](#) manual for details.

kerning true, false, \langle font set name \rangle 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 \XeTeX or \LuaTeX .

Table 1:

Availability of microtypographic 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 _Ǝ 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

pdfTeX 0.14f | LuaTeX 0.30 | XeTeX 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, $\langle dimension \rangle$	character
-------------	--	-----------

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

3.3 Font expansion

pdfTeX 0.14f | LuaTeX 0.30

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

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

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

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

stretch $\langle integer \rangle$ 20

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

step $\langle integer \rangle$ * 1

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

selected `true, false` false

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

3.4 Tracking

pdfTeX 1.40 | LuaTeX 0.62

letterspace $\langle integer \rangle$ 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.

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

3.6 Changing options later

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

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

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

4 Selecting fonts for micro-typography

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

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

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

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

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

If you now call

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

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

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

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

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 = <encoding>/<family>/<series>/<shape>/<size>'. This allows you to add fonts to the set that are otherwise disjunct from it. For instance, if you wanted to have the roman family in all sizes protruded, but only the normal sized, possibly italic, typewriter font (in contrast to, say, the small one), this is how you could declare the set:

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

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

Table 2 lists the 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 [<features>] {<set name>}
```

This command activates a font set previously declared by \DeclareMicrotypeSet. Using the optional argument, you can limit the application of the set to one or

more features. This command only has an effect if the feature was activated in the package options.

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

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

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

5 Micro fine tuning

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

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

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_l`). 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 ($=\{\langle left \rangle, \langle right \rangle\}$), possibly scaled by a factor. A unit setting will only be taken into account if it is not `=character`.

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

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

5.2 Font expansion

pdfTeX 0.14f | LuaTeX 0.30

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

By default, all characters of a font are allowed to be stretched or shrunk by the same amount. However, it is also possible to limit the expansion of certain characters if they are more sensitive to deformation. This is the purpose of the `\SetExpansion` command. Note that it will only have an effect if the package was 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 package was loaded with the selected option, 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 $\langle character \rangle = \langle expansion factor \rangle$ pairs. You may specify one number for each character, which determines the amount that a character may be expanded. The numbers denominate thousandths of the full expansion. For example, if you set the expansion factor for the character 'O' to 500, it will only be expanded or shrunk by one half of the amount that the rest of the characters will be expanded or shrunk. While the default value for character protrusion is 0 – that is, if you didn't specify any characters, none would be protruded – the default value for expansion is 1000, which means that all characters would be expanded by the same amount.

Options:

name, **load**, **preset**, **inputenc**, **context** Analogous to `\SetProtrusion`, the optional argument may be used to assign a name to the list, to load another list, to preset all expansion factors, to set the input encoding, or to determine the context of the list (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 1em (or the given unit); negative values are allowed, too.

Options:

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

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

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

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

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

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

no ligatures 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 is the (typographically dubious) outcome:

Stop stealing sheep!

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

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

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.

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

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

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

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

`\LoadMicrotypeFile` {}

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

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 \LaTeX , 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>}
```

`\textls*` While the tracking feature, described in section 5.3, will apply to sets of fonts, you may also want to letterspace shorter pieces of text, regardless of the font in which they are typeset.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 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. 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

pdf \TeX 1.30 | Lua \TeX 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 Lua \TeX , 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 `pdfcprot`). 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 \XeTeX and \LuaTeX). 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`.
- 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.
- Before this package was fully compatible with \LuaTeX , the following method of enabling expansion and protrusion with the `fontspec` package was most often recommended on the web and in groups and forums:

```
\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

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.

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

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.

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](#) and [Thành 2008](#) and elsewhere. I also thank him and the rest of the pdf \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 Bühmann* has suggested the possibility to specify ranges of font sizes, and resourcefully assisted in implementing this. He also came up with some good ideas for the management of complex configurations. *Ulrich Dirr* has made numerous suggestion, especially concerning the new extensions of interword spacing adjustment and additional character kerning. *Georg Duffner* has patiently tested `microtype` under $\X\TeX$ 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. *Loren B. Davis* has provided protrusion settings for OpenType versions of Palatino Linotype. I am also indebted to *Élie Roux*, who contributed the `lua` module.

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

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

¹⁴ Available from CTAN at [/fonts/ebgaramond](#), including configuration files for `microtype`.

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 and RazorXsr.

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/](#); 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](#))

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

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

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

Will Robertson, Khaled Hosny, *The fontspec package*, 6 May 2012. (Available from CTAN at [/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 and xunicode packages, viz. for OpenType fonts with Lua \TeX and Xe \TeX

- Support for protrusion with $X_{\text{d}}\text{T}_{\text{E}}\text{X} \geq 0.9997$
- Support for tracking/letterspacing with $\text{LuaT}_{\text{E}}\text{X} \geq 0.62$
- Allow context-sensitive setup with $\text{LuaT}_{\text{E}}\text{X}$
- Warning 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 $\text{pdfT}_{\text{E}}\text{X} \geq 1.40$ [3.3]

2.3c (2008/11/11)

- Support for $\text{LuaT}_{\text{E}}\text{X}$ enabled by default

2.3 (2007/12/23)

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

2.1 (2007/01/21)

- New command $\backslash\text{slig}$ to protect ligatures in letterspaced text [7]

2.0 (2007/01/14)

- Support for the new extensions of $\text{pdfT}_{\text{E}}\text{X} \geq 1.40$: tracking/letterspacing, adjustment of interword spacing (glue), and additional kerning (new commands $\backslash\text{SetTracking}$, $\backslash\text{SetExtraSpacing}$, $\backslash\text{SetExtraKerning}$; new options ‘tracking’, ‘spacing’, ‘kerning’) [5.3, 5.5, 5.4]
- New commands $\backslash\text{textls}$ and $\backslash\text{sstyle}$ 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 (pdf \TeX \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 (pdf \TeX \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
                  censo: 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*\lslig[1]{#1}
55 <package>
56 }

```

These commands also have a starred version.

```

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

```

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

```

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

```

Don't load letterspace.

```

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

```

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

```

66 \def\MT@old@cmd#1#2{%
67   \newcommand*#1{\MT@warning{%
68     \string#1 is deprecated. Please use\MessageBreak
69     \string#2 instead}%
70   \let #1#2#2}}

71 \MT@old@cmd\DeclareMicroTypeAlias\DeclareMicrotypeAlias
72 \MT@old@cmd\DeclareMicroTypeSet \DeclareMicrotypeSet
73 \MT@old@cmd\UseMicroTypeSet \UseMicrotypeSet
74 \MT@old@cmd\LoadMicroTypeFile \LoadMicrotypeFile
75 </package>

```

`\MT@warning` Communicate.

```

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

```

14.1.1 Debugging

`\tracingmicrotype` Cases for `\tracingmicrotype`:

```

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

```

```

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

```

```

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

```

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

Cases for `\tracingmicrotypeinpdf`:

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

```
98 \newcount\tracingmicrotypeinpdf
```

Let's see how it works . . .

```
\tracingmicrotypeinpdf=2
```

`\MT@pdf@annot` During font setup, we save the text for the popup in `\MT@pdf@annot`. (This requires $\text{pdf}_{\text{TeX}} \geq 1.30$.) The `pdfTeXcmds` package provides pdf_{TeX} 's utility commands in `\MT@addto@annot` pdf_{TeX} , too.

```

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

```

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

```
105 \newif\iftracingmicrotypeinpdfall
```

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

```

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

```

14.1.2 Requirements

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

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

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

```

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

```

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

```

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

```

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

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

```
\MT@engine@toold 173 \let\MT@engine\relax
174 <letterspace>\def\MT@engine@toold{0}
175 \ifx\pdfptextversion\@undefined \else
176   \ifx\pdfptextversion\relax \else
177     \def\MT@engine{pdf}
178     <letterspace> \let\MT@pdf@or@lua\@firstoftwo
179     <letterspace> \ifnum\pdfptextversion > 139 \def\MT@engine@toold{1}\fi
180     \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@pdfptext@no pdfT_EX's features for which we provide an interface here haven't always been available, and some specifics have changed over time. Therefore, we have to test which pdfT_EX we're using, if any. \MT@pdfptext@no will be used throughout the package to respectively do the right thing.

Currently, we have to distinguish seven cases for pdfT_EX:

- 0: not running pdfT_EX
- 1: pdfT_EX (< 0.14f)
- 2: + micro-typographic extensions (0.14f,g)
- 3: + protrusion relative to 1em (\geq 0.14h)
- 4: + automatic font expansion; protrusion no longer has to be set up first; scale factor fixed to 1000; default \efcode = 1000 (\geq 1.20)
- 5: + \(\left, right)marginkern; \pdfnoligatures; \pdfstrcmp; \pdfescapestring (\geq 1.30)
- 6: + adjustment of interword spacing; extra kerning; \letterspacefont; \pdfmatch¹⁵; \pdftracingfonts; always e-T_EX (\geq 1.40)
- 7: + \letterspacefont doesn't disable ligatures and kerns; \pdfcopyfont (\geq 1.40.4)

```
199 <*pdfptext-def>
200 <debug>\MT@edinfo@n1{0}{this is pdfptext \the\pdfptextversion(\pdfptextrevision)}
201 \def\MT@pdfptext@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@nl{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@nl{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@nl{0}{xetex no.: \MT@xetex@no}
239 </xetex-def>

```

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

0: N/A

1: Lua_{TeX} (< 0.36)

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

3: + \letterspacefont (≥ 0.62).

```

240 <*luatex-def>
241 <debug>\MT@info@nl{0}{this is luatex (\the\luatexversion)}

```

\MT@lua Communicate with lua. Beginning with Lua_{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@nl{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@tooold=\z@
257 \MT@warning@nl{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
                284 \newif\ifMT@auto
\ifMT@selected  285 \newif\ifMT@selected
\ifMT@noligatures 286 \newif\ifMT@noligatures
\ifMT@draft     287 \newif\ifMT@draft
                288 \newif\ifMT@spacing
\ifMT@spacing   289 \newif\ifMT@kerning
\ifMT@kerning   290 \newif\ifMT@tracking
\ifMT@tracking  291 \newif\ifMT@babel

```

`\MT@pr@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

```



```

\MT@pr@unit      Default unit for protrusion settings is character width, for spacing space, for kerning
\MT@sp@unit      (and tracking) 1 em.
\MT@kn@unit      298 \let\MT@pr@unit\@empty
                  299 \let\MT@sp@unit\m@ne
                  300 \def\MT@kn@unit{1em}

\MT@stretch      Expansion settings.
\MT@shrink      301 \let\MT@stretch\m@ne
\MT@step        302 \let\MT@shrink \m@ne
                  303 \let\MT@step \m@ne

\MT@pr@min      Minimum and maximum values allowed by pdfTeX.
\MT@pr@max      304 \def\MT@pr@min{-\@m}
\MT@ex@min      305 \let\MT@pr@max\@m
\MT@ex@max      306 \let\MT@ex@min\z@
\MT@ex@min      307 \let\MT@ex@max\@m
\MT@sp@min      308 \def\MT@sp@min{-\@m}
\MT@sp@max      309 \let\MT@sp@max\@m
\MT@kn@min      310 \def\MT@kn@min{-\@m}
\MT@kn@max      311 \let\MT@kn@max\@m
\MT@tr@min      312 </package>
\MT@tr@max      313 \def\MT@tr@min{-\@m}
\MT@tr@max      314 \let\MT@tr@max\@m
\MT@tr@max      315 <*package>

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

\MT@stretch@default Default values for expansion.
\MT@shrink@default 317 \def\MT@stretch@default{20 }
                  318 \def\MT@shrink@default{20 }

\MT@letterspace Default value for letterspacing (in thousandths of 1 em).
\MT@letterspace@default 319 </package>
                  320 \let\MT@letterspace\m@ne
                  321 \def\MT@letterspace@default{100}
                  322 <*package>

\ifMT@document Our private test whether we're still in the preamble.
                  323 \newif\ifMT@document
                  324 </package>
                  325 </package|letterspace>

```

14.1.4 Auxiliary macros

```

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

```

Some functions are loaded from a dedicated lua file. This avoids character escaping problems and incompatibilities between versions of LuaTeX. If available, we'll use the `luatextra` package to load the module.

```

338 <*/luatex-def>
339 \MT@lua{
340   if (luatextra and luatextra.use_module) then
341     luatextra.use_module("microtype")
342   else
343     dofile(kpse.find_file("microtype.lua"))
344   end}
345 </luatex-def>

```

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

```

346 <*/luafile>
347 if microtype then
348   -- we simply don't load
349 else
350
351 microtype = {}
352
353 microtype.module = {
354   name       = "microtype",
355   version    = 2.5,
356   date       = "2013/03/13",
357   description = "microtype module.",
358   author     = "Elie Roux & R Schlicht",
359   copyright  = "Elie Roux & R Schlicht",
360   license    = "LPPL",
361 }
362
363 if luatextra and luatextra.provides_module then
364   luatextra.provides_module(microtype.module)
365 end
366
367 </luafile>

```

To be continued, but first back to primitives.

\MT@glet Here's the forgotten one.

```

368 <*/package|letterspace>
369 \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.

```

370 \def\MT@exp@cs#1#2{\expandafter#1\csname#2\endcsname}
371 <*/package>
372 \def\MT@exp@gcs#1#2{\begingroup\expandafter\endgroup\expandafter#1\csname#2\endcsname}

```

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

```

373 \def\MT@def@n{\MT@exp@cs\def}
374 \def\MT@gdef@n{\MT@exp@gcs\gdef}

```

\MT@edef@n Its expanding versions.

```

375 </package>
376 \def\MT@edef@n{\MT@exp@cs\edef}
377 <*/package>
378 \def\MT@xdef@n{\MT@exp@gcs\xdef}

```

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

```

379 \def\MT@let@nc{\MT@exp@cs\let}
380 \def\MT@glet@nc{\MT@exp@gcs\MT@glet}

```

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

```

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

```

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

```

382 \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
383 \def\MT@glet@nn{\MT@exp@gcs{\global\expandafter\MT@let@cn}}

```

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

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

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

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

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

```
387 \def\MT@exp@two@c#1{\expandafter\expandafter\expandafter#1\expandafter}
388 <*package>
```

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

```
389 \def\MT@exp@two@n#1#2#3{%
390   \expandafter\expandafter\expandafter
391   #1\expandafter\expandafter\expandafter
392   {\expandafter#2\expandafter}\expandafter{#3}}
```

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

`\MT@ifdefined@c@T` Wrapper for testing whether command resp. `\csname` sequence is defined. If we

`\MT@ifdefined@c@TF` are running e-TeX, we will use its primitives `\ifdefined` and `\ifcsname`, which

`\MT@ifdefined@n@T` decreases memory use substantially.

```

\MT@ifdefined@n@TF 393 \def\MT@ifdefined@c@T#1{%
394   ^^X \ifdefined#1\expandafter\@firstofone\else\expandafter\@gobble\fi
395   ^^Q \ifx#1\@undefined\expandafter\@gobble\else\expandafter\@firstofone\fi
396 }
397 </package>
398 \def\MT@ifdefined@c@TF#1{%
399   ^^X \ifdefined#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
400 <package>^^Q \ifx#1\@undefined
401 <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
402 }
403 \def\MT@ifdefined@n@T#1{%
404   ^^X \ifcsname#1\endcsname\expandafter\@firstofone\else\expandafter\@gobble\fi
405 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
406 <package>^^Q \expandafter\@gobble\else\expandafter\@firstofone\fi
407 }
408 <*package>
409 \def\MT@ifdefined@n@TF#1{%
410   ^^X \ifcsname#1\endcsname\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
411   ^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
412   ^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
413 }

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

414 \def\MT@detokenize@n#1{%
415   ^^X \expandafter\MT@rem@last@space\detokenize{#1} \@nil
416   ^^Q \string#1%
417 }
418 \def\MT@detokenize@c#1{%
419   ^^X \MT@exp@one@n\MT@detokenize@n#1%
420   ^^Q \MT@exp@two@c\MT@rem@last@space\strip@prefix\meaning#1 \@nil
421 }
422 \def\MT@rem@last@space#1 #2{#1%
423   \ifx\@nil#2\else \space
424   \expandafter\MT@rem@last@space\expandafter#2\fi
425 }

\MT@ifempty Test whether argument is empty.

426 </package>
427 \begingroup
428 \catcode`\%=12
```

```

429 \catcode`\&=14
430 \gdef\MT@ifempty#1{&
431   \if %#1%&
432     \expandafter\@firstoftwo
433   \else
434     \expandafter\@secondoftwo
435   \fi
436 }
437 \endgroup
438 (*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).

439 (/package)
440 (/package|letterspace)
441 (pdfTeX-def)\MT@requires@pdfTeX6{
442 (letterspace)\MT@pdf@or@lua{
443 (*pdfTeX-def|letterspace)
444 \def\MT@ifint#1{%
445   \ifcase\pdfmatch{^-*[0-9]+ *$}{#1}\relax
446     \expandafter\@secondoftwo
447   \else
448     \expandafter\@firstoftwo
449   \fi
450 }
451 }{
452 (/pdfTeX-def|letterspace)
453 (*pdfTeX-def|xetex-def|letterspace)
454 \def\MT@ifint#1{%
455   \if!\ifnum9<1#1!\else?\fi
456     \expandafter\@firstoftwo
457   \else
458     \expandafter\@secondoftwo
459   \fi
460 }
461 (/pdfTeX-def|xetex-def|letterspace)
462 (pdfTeX-def|letterspace)}
463 (luatex-def)\def\MT@ifint#1{\csname\MT@lua{microtype.if_int[[#1]]}\endcsname}
464 (*luafile)
465 function microtype.if_int(s)
466   if string.find(s,"^-*[0-9]+ *$") then
467     tex.write("@firstoftwo")
468   else
469     tex.write("@secondoftwo")
470   end
471 end
472
473 (/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.)

474 (*pdfTeX-def)
475 \MT@requires@pdfTeX6{
476 \def\MT@ifdimen#1{%
477   \ifcase\pdfmatch{^([0-9]+([.],[0-9]+)?|[.],[0-9]+)%
478     (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
479     \expandafter\@secondoftwo
480   \else
481     \expandafter\@firstoftwo
482   \fi
483 }
484 }{
485 (pdfTeX-def)

```

```

486 <*pdftex-def>|xetex-def>
487 \def\MT@ifdimen#1{%
488   \setbox\z@=\hbox{%
489     \MT@count=1#1\relax
490     \ifnum\MT@count=\@ne
491       \aftergroup\@secondoftwo
492     \else
493       \aftergroup\@firstoftwo
494     \fi
495   }%
496 }
497 </pdftex-def>|xetex-def>
498 <pdftex-def>
499 <luatex-def>\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen}[[#1]]\endcsname}
500 <*luafile>
501 function microtype.if_dimen(s)
502   if (string.find(s, "^-[0-9]+(%a*) *$") or
503       string.find(s, "^-[0-9]*[.][0-9]+(%a*) *$")) then
504     tex.write("@firstoftwo")
505   else
506     tex.write("@secondoftwo")
507   end
508 end
509
510 </luafile>

\MT@ifdim    Test floating point numbers.

511 <*package>
512 \def\MT@ifdim#1#2#3{%
513   \ifdim #1\p@ #2 #3\p@
514     \expandafter\@firstoftwo
515   \else
516     \expandafter\@secondoftwo
517   \fi
518 }
519 </package>

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

520 <*pdftex-def>
521 \MT@requires@pdftex5{
522 \def\MT@ifstreq#1#2{%
523   \ifcase\pdfstrcmp{#1}{#2}\relax
524     \expandafter\@firstoftwo
525   \else
526     \expandafter\@secondoftwo
527   \fi
528 }
529 }{
530 </pdftex-def>
531 <*pdftex-def>|xetex-def>
532 \def\MT@ifstreq#1#2{%
533   \edef\MT@resa{#1}%
534   \edef\MT@resb{#2}%
535   \ifx\MT@resa\MT@resb
536     \expandafter\@firstoftwo
537   \else
538     \expandafter\@secondoftwo
539   \fi
540 }
541 </pdftex-def>|xetex-def>
542 <pdftex-def>
543 <luatex-def>\def\MT@ifstreq#1#2{\csname\MT@lua{microtype.if_str_eq}[[#1]], [[#2]]\endcsname}
544 <*luafile>
545 function microtype.if_str_eq(s1, s2)
546   if s1 == s2 then

```

```

547     tex.write("@firstoftwo")
548   else
549     tex.write("@secondoftwo")
550   end
551 end
552
553 </luafile>

\MT@xadd    Add item to a list.
554 <*package>
555 \def\MT@xadd#1#2{%
556   \ifx#1\relax
557     \xdef#1{#2}%
558   \else
559     \xdef#1{#1#2}%
560   \fi
561 }

\MT@xaddb   Add item to the beginning.
562 \def\MT@xaddb#1#2{%
563   \ifx#1\relax
564     \xdef#1{#2}%
565   \else
566     \xdef#1{#2#1}%
567   \fi
568 }
569 </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@    570 <*package|letterspace>
\MT@clist@function 571 \def\MT@map@clist@n#1#2{%
\MT@clist@break    572   \ifx\@empty#1\else
                    573     \def\MT@clist@function##1{#2}%
                    574     \MT@map@clist@#1,\@nil,\@nnil
                    575   \fi
                    576 }
                    577 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}
                    578 \def\MT@map@clist@#1,{%
                    579   \ifx\@nil#1%
                    580     \expandafter\MT@clist@break
                    581   \fi
                    582   \MT@clist@function{#1}%
                    583   \MT@map@clist@
                    584 }
                    585 \let\MT@clist@function\@gobble
                    586 \def\MT@clist@break#1\@nnil{}
                    587 <*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@    588 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break    589 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
                    590 \def\MT@map@tlist@#1#2{%
                    591   \ifx\@nnil#2\else
                    592     #1{#2}%
                    593     \expandafter\MT@map@tlist@
                    594     \expandafter#1%
                    595   \fi
                    596 }
                    597 \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@    598 \newif\ifMT@inlist@

```

```

599 \def\MT@in@clist#1#2{%
600   \def\MT@res@a#1,#1,##2##3\@nnil{%
601     \ifx##2\@empty
602       \MT@inlist@false
603     \else
604       \MT@inlist@true
605     \fi
606   }%
607   \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil
608 }

```

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

```

609 \def\MT@rem@from@clist#1#2{%
610   \def\MT@res@a#1,#1,##2\MT@res@a{##1,##2\MT@res@b}%
611   \def\MT@res@b#1,\MT@res@b##2\MT@res@b{\ifx,##1\@empty\else##1\fi}%
612   \xdef#2{\MT@exp@two@c\MT@res@b\MT@res@a\expandafter,#2,\MT@res@b,#1,\MT@res@a}%
613 }

```

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

```

614 \def\MT@in@tlist#1#2{%
615   \MT@inlist@false
616   \def\MT@res@a{#1}%
617   \MT@map@tlist@c#2\MT@in@tlist@
618 }
619 \def\MT@in@tlist@#1{%
620   \edef\MT@res@b{#1}%
621   \ifx\MT@res@a\MT@res@b
622     \MT@inlist@true
623   \expandafter\MT@tlist@break
624   \fi
625 }

```

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

```

\MT@in@rlist@  \MT@size@name
\MT@in@rlist@@ 626 \def\MT@in@rlist#1{%
\MT@size@name 627   \MT@inlist@false
628   \MT@map@tlist@c#1\MT@in@rlist@
629 }
630 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
631 \def\MT@in@rlist@@#1#2#3{%
632   \MT@ifdim{#2}=\m@ne{%
633     \MT@ifdim{#1}=\MT@size
634     \MT@inlist@true
635     \relax
636   }{%
637     \MT@ifdim\MT@size<{#1}\relax{%
638       \MT@ifdim\MT@size<{#2}%
639       \MT@inlist@true
640       \relax
641     }%
642   }%
643   \ifMT@inlist@
644     \def\MT@size@name{#3}%
645     \expandafter\MT@tlist@break
646   \fi
647 }

```

`\MT@loop` This is the same as L^AT_EX's `\loop`, which we mustn't use, since this could confuse an outer `\loop` in the document.

`\MT@repeat` 648 */package*
649 `\def\MT@loop#1\MT@repeat{%`

```

650 \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
651 \MT@iterate \let\MT@iterate\relax
652 }
653 \let\MT@repeat\fi

\MT@while@num    Execute <#3> from <#1> up to (excluding) <#2> (much faster than LATEX's \@whilenum).
654 \def\MT@while@num#1#2#3{%
655   \@tempcnta#1\relax
656   \MT@loop #3%
657   \advance\@tempcnta \@ne
658   \ifnum\@tempcnta < #2\MT@repeat
659 }

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

\MT@count        Increment macro <#1> by one. Saves using up too many counters. The e-TEX way is
\MT@increment     slightly faster.
676 \newcount\MT@count
677 \def\MT@increment#1{%
678   ^^X \edef#1{\number\numexpr #1 + 1\relax}%
679   ^^Q \MT@count=#1\relax
680   ^^Q \advance\MT@count \@ne
681   ^^Q \edef#1{\number\MT@count}%
682 }

\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.
683 \def\MT@scale#1#2#3{%
684   ^^Q \multiply #1 #2\relax
685   \ifnum #3 = \z@
686     ^^X #1=\numexpr #1 * #2\relax
687   \else
688     ^^X #1=\numexpr #1 * #2 / #3\relax
689     ^^Q \divide #1 #3\relax
690   \fi
691 }

\MT@abbr@pr      Some abbreviations. Thus, we can have short command names but full-length log
\MT@abbr@ex      output.
\MT@abbr@pr@c    692 \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

```



```

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

```

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

```

\MT@rbba@expansion 707 \def\MT@rbba@protrusion{pr}
\MT@rbba@spacing    708 \def\MT@rbba@expansion{ex}
\MT@rbba@kerling    709 \def\MT@rbba@spacing{sp}
\MT@rbba@tracking   710 \def\MT@rbba@kerling{kn}
\MT@rbba@tracking   711 \def\MT@rbba@tracking{tr}

```

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

```

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

```

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

```

714 \def\MT@is@feature#1{%
715   \MT@exp@one@n\MT@in@clist\@tempa\MT@features@long
716   \ifMT@inlist@
717     \expandafter\@firstofone
718   \else
719     \MT@error{\@tempa' is not an available micro-typographic\MessageBreak
720       feature. Ignoring #1}{Available features are: \MT@features@long'.}%
721     \expandafter\@gobble
722   \fi
723 }

```

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.

```

724 \@ifl@aded{tex}{wordcount}{%
725   \MT@warning@nl{Detected the `wordcount' utility.\MessageBreak
726     Disabling \MT@MT', since it wouldn't work}%
727   \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.

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

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

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

Don't hesitate with `miniltx`.

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

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

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

`\MT@with@babel@and@T` \LaTeX 's `\ifpackage@with` ignores the class options.

```
737 \def\MT@with@babel@and@T#1{%
738 \MT@ifdefined@n@T{opt@babel.\@pkgextension}{%
739 \expandafter\MT@in@clist{#1}
740 {\csname opt@babel.\@pkgextension\endcsname,\@classoptionslist}%
741 \ifMT@in@list\expandafter\@gobble\fi
742 }@gobble
743 }
```

`\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 $\text{pdf}\text{\TeX}$ 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 $\text{pdf}\text{\TeX}$ 1.21b (aka. 1.30.0). They are also part of recent $\text{X}\text{\TeX}$.

```
744 </package>
745 <pdfTeX-def>\MT@requires@pdfTeX5{
746 <*pdfTeX-def|luatex-def|xetex-def>
747 \def\MT@ledmac@setup{%
748 \ifMT@protrusion
749 \MT@ifdefined@c@TF\l@dunhbox@line{%
```

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

```
\MT@led@kern 750 \MT@info@n1{Patching ledmac to enable character protrusion}%
751 \newdimen\MT@led@kern
752 \let\MT@led@unhbox@line\l@dunhbox@line
753 \renewcommand*{\l@dunhbox@line}[1]{%
754 \ifhbox#1%
755 \MT@led@kern=\rightmarginkern##1%
756 \kern\leftmarginkern##1%
757 \MT@led@unhbox@line##1%
758 \kern\MT@led@kern
759 \fi
760 }%
761 }{%
762 \MT@warning@n1{%
763 Character protrusion in paragraphs with line\MessageBreak
764 numbering will only work if you update ledmac}%
765 }%
766 \fi
```

```

767 }
768 </pdfTeX-def|luatex-def|xetex-def>
769 <*pdfTeX-def>
770 }{
771   \def\MT@ledmac@setup{%
772     \ifMT@protrusion
773       \MT@warningonl{%
774         The pdfTeX version you are using does not allow\MessageBreak
775         character protrusion in paragraphs with line\MessageBreak
776         numbering by the 'ledmac' package.\MessageBreak
777         Upgrade pdfTeX to version 1.30 or later}%
778       \fi
779     }
780 }
781 </pdfTeX-def>

```

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

```

782 <*package|letterspace>
783 <*package>
784 \def\MT@restore@p@h{\chardef\%~\% \chardef\#~\# }

```

Two new conditionals for use with Xe_{La}TeX or Lua_{La}TeX.

\ifMT@unicode

```

\ifMT@fontspec 785 \newif\ifMT@unicode
786 \newif\ifMT@fontspec
787 \MT@with@package@T{xunicode}\MT@xunicodetrue
788 \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.)

```

789 \def\MT@setupfont@hook{%

```

When a font is defined via \fontspec, the font is not actually loaded, hence Xe_{La}TeX resp. Lua_{La}TeX would see a wrong font (in \MT@get@slot). Therefore, we load the current font.

```

790 \ifMT@fontspec\MT@font\fi

```

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

```

791 \MT@if@false
792 \MT@with@babel@and@T{spanish} \MT@if@true
793 \MT@with@babel@and@T{galician}\MT@if@true
794 \MT@with@babel@and@T{mexican} \MT@if@true
795 \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.)

```

796 \MT@with@package@T{csquotes}{%
797   \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.

```

798 \MT@if@false

```

```

799 \MT@with@package@T{hyperref} \MT@if@true
800 \MT@with@package@T{tex4ht} \MT@if@true
801 \MT@with@package@T{mathastext}\MT@if@true
802 \ifMT@if@MT@restore@p@h\fi
803 }

```

Check again at the end of the preamble.

```

804 </package>
805 \MT@addto@setup{%
806 <*package>

```

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

```

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

```

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

```

817 \let\MT@setupfont@hook\@empty
818 \ifMT@fontspec
819 \g@addto@macro\MT@setupfont@hook{\MT@font}%
820 \fi
821 \MT@if@false
822 \MT@with@babel@and@T{spanish} \MT@if@true
823 \MT@with@babel@and@T{galician}\MT@if@true
824 \MT@with@babel@and@T{mexican} \MT@if@true
825 \ifMT@if@
826 \g@addto@macro\MT@setupfont@hook{%
827 \MT@ifdefined@c@T\percentsign{\let%\percentsign}}%
828 \fi
829 \MT@with@package@T{csquotes}{%
830 \ifpackage@later{csquotes}{2005/05/11}{%
831 \g@addto@macro\MT@setupfont@hook{\@disablequotes
832 }{%
833 \MT@warning@n1{%
834 Should you receive warnings about unknown slot\MessageBreak
835 numbers, try upgrading the `csquotes' package}%
836 }%
837 }%

```

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

```

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

```

```

854 <*package>
855 \MT@with@package@T{tex4ht}\MT@if@true
856 \MT@with@package@T{mathastext}\MT@if@true
857 \ifMT@if@true\g@addto@macro\MT@setupfont@hook\MT@restore@p@h\fi

```

The listings package makes numbers and letters active,

```

858 \MT@with@package@T{listings}{%
859 \g@addto@macro\MT@cfg@catcodes{%
860 \MT@while@num{"30"}{"3A"}{\catcode\@tempcnta 12\relax}%
861 \MT@while@num{"41"}{"5B"}{\catcode\@tempcnta 11\relax}%
862 \MT@while@num{"61"}{"7B"}{\catcode\@tempcnta 11\relax}%
863 }%

```

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

```

864 \g@addto@macro\MT@setupfont@hook{%
865 \catcode`\z@

```

Inside a listing, \space is redefined.

```

866 \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.

```

867 \let\lst@ProcessLetter\empty
868 }%
869 }%

```

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.

```

870 </package>
871 <plain> \MT@requires@latex2{
872 \MT@with@package@T{soul}{%
873 \soulregister\lsstyle 0%
874 \soulregister\textls 1%
875 }%

```

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.

```

876 <*plain>
877 }{\ifx\SOUL@\undefined\else
878 \soulregister\lsstyle 0%
879 \soulregister\textls 1%
880 \fi}%
881 </plain>
882 <*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.

```

883 \MT@with@package@T{pinyin}{%
884 \let\MT@orig@py@macron\py@macron
885 \ifpackageafter{pinyin}{2005/08/11}{% 4.6.0
886 \def\py@macron#1#2{%
887 \let\pickup@font\MT@orig@pickupfont
888 \MT@orig@py@macron{#1}{#2}%
889 \let\pickup@font\MT@pickupfont}%
890 }{%
891 \def\py@macron#1{%
892 \let\pickup@font\MT@orig@pickupfont
893 \MT@orig@py@macron{#1}%
894 \let\pickup@font\MT@pickupfont}%
895 }%

```

```

896 }%
897 </package>
898 }
899 </package|letterspace>

```

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

```

900 <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.

```

901 <*pdfTeX-def|xetex-def|luatex-def>
902 \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).

```

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

```

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

```

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

```

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

```

908   \MT@exp@one@n\MT@find@file\MT@family
909   \ifx\MT@familyalias\@empty \else
910     \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.)

```

911 % \ifx\f@encoding\cf@encoding\else\@enc@update\fi
912 }

```

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

```

913 <pdfTeX-def>\MT@requires@pdfTeX6
914 <luatex-def>\MT@requires@luatex3
915 <pdfTeX-def|luatex-def> {\g@addto@macro\MT@setupfont\MT@tracking}\relax
916 \g@addto@macro\MT@setupfont{%
917   \MT@check@font
918   \ifMT@inlist@
919 <debug>\MT@show@pdfannot2%
920   \else
921     \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.

```

922   \MT@protrusion
923 <pdfTeX-def|luatex-def> \MT@expansion
924 }

```

Interword spacing and kerning (pdfTeX 1.40).

```
925 <pdfTex-def>
926 \MT@requires@pdfTeX6{
927 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
928 }\relax
929 </pdfTex-def>
```

Disable ligatures (pdfTeX 1.30).

```
930 <pdfTex-def>\MT@requires@pdfTeX5{
931 <pdfTex-def>\luatex-def\g@addto@macro\MT@setupfont\MT@noligatures
932 <pdfTex-def>}\relax
933 \g@addto@macro\MT@setupfont{%
```

Debugging.

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

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

```
935 \MT@register@font
936 \fi
937 }
938 </pdfTex-def>\xetex-def\luatex-def
```

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

```
939 <pdfTex-def>\luatex-def
940 \let\MT@copy@font\relax
941 <pdfTex-def>\MT@requires@pdfTeX7{
942 \def\MT@copy@font{%
```

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

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

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

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

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

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

We only need the font identifier for letterspacing.

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

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

```
962 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
963 }
```

\MT@rem@from@list

```

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

```

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

```

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

```

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

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

\MT@encoding
\MT@family

```

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

```

\MT@familyalias Alias family?

```

983 \MT@ifdefined@n@TF{MT@\MT@family @alias}%
984 { \MT@let@cn\MT@familyalias{MT@\MT@family @alias}}%
985 { \let\MT@familyalias\empty}%
986 }

```

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

```

\MT@scrubfeatures 987 \def\MT@scrubfeature#1(#2)#3\relax{#1}
988 \def\MT@scrubfeatures#1(#2)#3\relax{%
989   #1%
990   \ifx\relax#3\relax\else

```



```

991     \MT@scrubfeatures#3\relax
992     \fi
993 }

\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 994 \newif\ifMT@do
995 \def\MT@maybe@do#1{%
    (but only if the feature isn't globally set to false)
996     \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).

997     \MT@dotrue
998     \edef\@tempa{\csname MT@#1\setname\endcsname}%
999     \MT@map@clist@n{font,encoding,family,series,shape,size}{%
1000         \MT@ifdefined@n@TF{MT@checklist@#1}%
1001         {\csname MT@checklist@#1\endcsname}%
1002         {\MT@checklist@{#1}}%
1003         {#1}%
1004     }%
1005     \else
1006         \MT@dofalse
1007     \fi
1008     \ifMT@do

    \MT@feat stores the current feature.

1009     \def\MT@feat{#1}%
1010     \csname MT@set@#1\codes\endcsname
1011     \else
1012         \MT@vinfo{... No \@nameuse{MT@abbr@#1}}%
1013     \fi
1014 }

\MT@dinfo@list

1015 <debug>\def\MT@dinfo@list#1#2#3{\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: #2
1016 <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).

1017 \def\MT@checklist@#1#2{%
1018 <debug> \MT@ifdefined@n@T
1019 <debug> \MT@ifdefined@n@TF
1020     {MT@#2list@#1\@tempa}{%

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

1021     \expandafter\MT@exp@one@n\expandafter\MT@in@clist
1022     \csname MT@#1\expandafter\endcsname
1023     \csname MT@#2list@#1\@tempa\endcsname
1024     \ifMT@inlist@
1025 <debug>\MT@dinfo@list{#2}{#1}{in}%
1026     \MT@dotrue
1027     \else
1028 <debug>\MT@dinfo@list{#2}{#1}{not in}%
1029     \MT@dofalse
1030     \expandafter\MT@clist@break
1031     \fi
1032 }%

```

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.

```
1033 <debug> {\MT@info@list{#2}{#1}{}}%
1034 }
```

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

```
1035 \def\MT@checklist@family#1{%
1036 <!debug> \MT@ifdefined@n@T
1037 <debug> \MT@ifdefined@n@TF
1038   {MT@#1list@family@ \@tempa}%
1039   \MT@exp@two@n\MT@in@clist
1040   \MT@family{\csname MT@#1list@family@ \@tempa\endcsname}%
1041   \ifMT@inlist@
1042 <debug> \MT@info@list{#1}{family}{in}%
1043   \MT@dotrue
1044   \else
1045 <debug> \MT@info@list{#1}{family}{not in}%
1046   \MT@dofalse
1047   \ifx\MT@familyalias\@empty \else
1048     \MT@exp@two@n\MT@in@clist
1049     \MT@familyalias{\csname MT@#1list@family@ \@tempa\endcsname}%
1050     \ifMT@inlist@
1051 <debug> \MT@info@list{#1}{family alias}{in}%
1052     \MT@dotrue
1053 <debug> \else\MT@info@list{#1}{family alias}{not in}%
1054     \fi
1055   \fi
1056   \fi
1057   \ifMT@do \else
1058     \expandafter\MT@clist@break
1059   \fi
1060 }%
1061 <debug> {\MT@info@list{#1}{family}{}}%
1062 }
```

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

```
1063 \def\MT@checklist@size#1{%
1064 <!debug> \MT@ifdefined@n@T
1065 <debug> \MT@ifdefined@n@TF
1066   {MT@#1list@size@ \@tempa}%
1067   \MT@exp@cs\MT@in@rlist{MT@#1list@size@ \@tempa}%
1068   \ifMT@inlist@
1069 <debug> \MT@info@list{#1}{size}{in}%
1070   \MT@dotrue
1071   \else
1072 <debug> \MT@info@list{#1}{size}{not in}%
1073   \MT@dofalse
1074   \expandafter\MT@clist@break
1075   \fi
1076 }%
1077 <debug> {\MT@info@list{#1}{size}{}}%
1078 }
```

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

```
1079 \def\MT@checklist@font#1{%
1080 <!debug> \MT@ifdefined@n@T
1081 <debug> \MT@ifdefined@n@TF
1082   {MT@#1list@font@ \@tempa}%
```

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

```
1083   \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1084   \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter
1085   \@tempb \csname MT@#1list@font@ \@tempa\endcsname
1086   \ifMT@inlist@
1087 <debug> \MT@info@list{#1}{font}{in}%
1088   \expandafter\MT@clist@break
```

```

1089     \else
1090     <debug>\MT@info@list{#1}{font}{not in}%
1091     \MT@dofalse
1092     \fi
1093     }%
1094     <debug> {\MT@info@list{#1}{font}{}}%
1095 }

```

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`.

```

1096 \newif\ifMT@nofamily
1097 </package>

```

`\MT@protrusion` Set up for protrusion?

```

1098 <*pdfTeX-def|xetex-def|luatex-def>
1099 \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.

```

1100 \def\MT@set@pr@codes{%
1101   \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).

```

1102   \MT@if@list@exists{%
1103     \ifMT@nofamily
1104       \MT@ifdefined@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1105         \MT@info@n1{Loading generic settings for font family\MessageBreak
1106                     '\MT@family' (encoding: \MT@encoding).\MessageBreak
1107                     For optimal results, create family-specific settings.\MessageBreak
1108                     See the microtype manual for details}%
1109         \MT@gl@et@nc{\MT@encoding-\MT@family-settings}\@empty
1110       }%
1111     \fi
1112     \MT@get@font@dimen@six{%
1113       \MT@get@opt
1114       \MT@reset@pr@codes

```

Get the name of the inheritance list and parse it.

```

1115     \MT@get@inh@list

```

Set an input encoding?

```

1116     \MT@set@inputenc{c}%

```

Load additional lists?

```

1117     \MT@load@list\MT@pr@c@name
1118     \MT@set@listname

```

Load the main list.

```

1119     \MT@let@cn\@tempc{\MT@pr@c@\MT@pr@c@name}%
1120     \expandafter\MT@set@codes\@tempc,\relax,%
1121   }\MT@reset@pr@codes
1122 }

```

`\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`

```

1123 \def\MT@get@font@dimen@six{%
1124   \ifnum\fontdimen6\MT@font=\z@
1125     \MT@warning@n1{%

```

```

1126     Font '\MT@font' does not specify its\MessageBreak
1127     \@backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
1128     \@nameuse{MT@abbr@MT@feat} will not work with this font}%
1129     \expandafter\@gobble
1130   \else
1131     \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1132     \expandafter\@firstofone
1133   \fi
1134 }

\MT@set@all@pr    Set all protrusion codes of the font.
1135 \def\MT@set@all@pr#1#2{%
1136   <debug>\MT@info{n}{3}{-- lp/rp: setting all to #1/#2}%
1137   \let\MT@temp\@empty
1138   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
1139   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
1140   \MT@do@font\MT@temp
1141 }

\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.
1142 \def\MT@reset@pr@codes@{\MT@set@all@pr\z@\z@}
1143 \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.
1144 \def\MT@the@pr@code{\@tempcntb}
1145 <pdfTeX-def|LaTeX-def>
1146 <pdfTeX-def>\MT@requires@pdfTeX6
1147 <LaTeX-def>\MT@requires@LaTeX3
1148 {\def\MT@the@pr@code@tr{%
1149   \numexpr\@tempcntb+\MT@letterspace@/2\relax
1150 }
1151 }\relax
1152 </pdfTeX-def|LaTeX-def>

\MT@set@codes    Split up the values and set the codes.
1153 \def\MT@set@codes#1,{%
1154   \ifx\relax#1\@empty\else
1155     \MT@split@codes #1==\relax
1156     \expandafter\MT@set@codes
1157   \fi
1158 }

\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.
1159 \def\MT@split@codes#1=#2=#3\relax{%
1160   \def\@tempa{#1}%
1161   \ifx\@tempa\@empty \else
1162     \MT@get@slot
1163     <pdfTeX-def|LaTeX-def> \ifnum\MT@char > \m@ne
1164     <xetex-def> \ifx\MT@char\@empty \else
1165       \MT@get@char@unit
1166       \csname MT@\MT@feat @split@val\endcsname#2\relax
1167     \fi
1168   \fi
1169 }

\MT@pr@split@val
1170 \def\MT@pr@split@val#1,#2\relax{%
1171   \def\@tempb{#1}%

```

```

1172 \MT@ifempty\@tempb\relax{%
1173   \MT@scale@to@em
1174   \lrcode\MT@font\MT@char=\MT@the@pr@code
1175 (debug)\MT@info@nl{4}{;;; lp (\MT@char): \number\lrcode\MT@font\MT@char\space: [#1]}%
1176 }%
1177 \def\@tempb{#2}%
1178 \MT@ifempty\@tempb\relax{%
1179   \MT@scale@to@em
1180   \rprcode\MT@font\MT@char=\MT@the@pr@code
1181 (debug)\MT@info@nl{4}{;;; rp (\MT@char): \number\rprcode\MT@font\MT@char\space: [#2]}%
1182 }%

```

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

```

1183 \MT@ifdefined@c@T\MT@pr@inh@name{%
1184   \MT@ifdefined@n@T{\MT@inh@\MT@pr@inh@name @\MT@char @}{%
1185     \MT@exp@cs\MT@map@tlist@c
1186     {\MT@inh@\MT@pr@inh@name @\MT@char @}%
1187     \MT@set@pr@heirs
1188   }%
1189 }%
1190 }

```

`\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]pcode`’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`.

```

1191 (pdfTeX-def)\MT@requires@pdfTeX3{
1192 \def\MT@scale@to@em{%
1193   \@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`.

```

1194   \MT@scale\@tempcntb \@tempb \MT@dimen@six
1195   \ifnum\@tempcntb=\z@ \else
1196     \MT@scale@factor
1197   \fi
1198 }

```

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

```

1199 \def\MT@get@charwd{%
1200 (*pdfTeX-def)
1201 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1202 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1203 ^^Q \MT@count=\wd\z@
1204 (/pdfTeX-def)
1205 (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).

```

1206 (*xetex-def)
1207   \ifnum\MT@char@<\z@
1208     \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%

```

```

1209 \MT@count=\wd\z@
1210 \else
1211 \MT@count=\fontcharwd\MT@font\MT@char@\relax
1212 \fi
1213 /xetex-def
1214 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1215 }

```

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`.

```

1216 (*pdfTeX-def)
1217 \MT@requires@pdfTeX6{
1218 \g@addto@macro\MT@get@charwd{%
1219 \MT@ifdefined@ecT\MT@letterspace@
1220 {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1221 }
1222 }\relax
1223 }{

```

No adjustment with versions 0.14f and 0.14g.

```

1224 \def\MT@scale@to@em{%
1225 \MT@count=\@tempb\relax
1226 \ifnum\MT@count=\z@ \else
1227 \MT@scale@factor
1228 \fi
1229 }

```

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

```

1230 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1231 }
1232 /pdfTeX-def
1233 /pdfTeX-def|xetex-def|luatex-def

```

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

```

1234 (*package)
1235 \def\MT@get@font@dimen#1{%
1236 \ifnum\fontdimen#1\MT@font=\z@
1237 \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1238 \@backslashchar fontdimen #1 (it's zero)! \MessageBreak
1239 You should use a different 'unit' for \MT@curr@list@name}%
1240 \else
1241 \MT@count=\fontdimen#1\MT@font
1242 \fi
1243 }

```

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

```

1244 \def\MT@info@missing@char{%
1245 \MT@info@n1{Character '\the\MT@toks'
1246 ^^X \iffontchar\MT@font\MT@char@
1247 has a width of 0pt
1248 ^^X \else is missing\fi
1249 ^^Q \MessageBreak (it's probably missing)
1250 \MessageBreak in font '\MT@font'. \MessageBreak
1251 Ignoring protrusion settings for this character}%
1252 }

```

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

```

1253 \def\MT@scale@factor{%
1254 \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1255 \expandafter\MT@scale\expandafter \@tempcntb
1256 \csname MT@\MT@feat @factor@\endcsname \@m
1257 \fi
1258 \ifnum\@tempcntb>\csname MT@\MT@feat @max\endcsname\relax

```

```

1259 \MT@exp@cs\MT@warn@code@too@large{MT@MT@feat @max}%
1260 \else
1261 \ifnum\@tempcntb<\csname MT@MT@feat @min\endcsname\relax
1262 \MT@exp@cs\MT@warn@code@too@large{MT@MT@feat @min}%
1263 \fi
1264 \fi
1265 }

```

`\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.

```

1266 \def\MT@warn@code@too@large#1{%
1267 \@tempcnta=#1\relax
1268 \ifnum\csname MT@MT@feat @factor\endcsname=\@m \else
1269 \expandafter\MT@scale\expandafter\@tempcnta\expandafter
1270 \@m \csname MT@MT@feat @factor\endcsname
1271 \fi
1272 \MT@scale\@tempcnta \MT@dimen@six \MT@count
1273 \MT@warning@n1{The \@nameuse{MT@abbr@MT@feat} code \@temp\space
1274 is too large for character\MessageBreak
1275 \the\MT@toks' in \MT@curr@list@name.\MessageBreak
1276 Setting it to the maximum of \number\@tempcnta}%
1277 \@tempcntb=#1\relax
1278 }

```

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

```

1279 \def\MT@get@opt{%
1280 \MT@set@listname

```

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

```

\MT@sp@factor@ 1281 \MT@ifdefined@n@TF{MT@MT@feat @c@\csname MT@MT@feat @c@name\endcsname @factor}{%
\MT@kn@factor@ 1282 \MT@let@nn{MT@MT@feat @factor@}
1283 {MT@MT@feat @c@\csname MT@MT@feat @c@name\endcsname @factor}%
1284 \MT@vinfo{... : Multiplying \@nameuse{MT@abbr@MT@feat} codes by
1285 \number\csname MT@MT@feat @factor\endcsname/1000}%
1286 }{%
1287 \MT@let@nn{MT@MT@feat @factor@}{MT@MT@feat @factor}%
1288 }%

```

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

```

\MT@kn@unit@ 1289 \MT@ifdefined@n@TF{MT@MT@feat @c@\csname MT@MT@feat @c@name\endcsname @unit}{%
1290 \MT@let@nn{MT@MT@feat @unit@}%
1291 {MT@MT@feat @c@\csname MT@MT@feat @c@name\endcsname @unit}%
1292 \MT@exp@cs\ifx{MT@MT@feat @unit@}\@empty
1293 \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} codes
1294 relative to character widths}%
1295 \else
1296 \MT@exp@cs\ifx{MT@MT@feat @unit@}\m@ne
1297 \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} codes
1298 relative to width of space}%
1299 \fi
1300 \fi
1301 }{%
1302 \MT@let@nn{MT@MT@feat @unit@}{MT@MT@feat @unit}%
1303 }%

```

`\MT@get@space@unit` The codes are either relative to character widths, or to a fixed width. For spacing
`\MT@get@char@unit` 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.

```

1304 \let\MT@get@char@unit\relax
1305 \let\MT@get@space@unit@gobble

```

```

1306 \MT@exp@cs\ifx{MT@MT@feat @unit@}\@empty
1307 \let\MT@get@char@unit\MT@get@charwd
1308 \else
1309 \MT@exp@cs\ifx{MT@MT@feat @unit@}\m@ne
1310 \let\MT@get@space@unit\MT@get@font@dimen
1311 \else
1312 \MT@exp@cs\MT@get@unit{MT@MT@feat @unit@}%
1313 \fi
1314 \fi

```

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

```

1315 \MT@ifdefined@n@T{MT@MT@feat @c@\csname MT@MT@feat @c@name\endcsname @preset}{%
1316 \csname MT@preset@MT@feat\endcsname
1317 \MT@let@nc{MT@reset@MT@feat @codes}\relax
1318 }%
1319 }

```

\MT@get@unit If unit contains an em or ex, we use the corresponding \fontdimen to obtain the real size. Simply converting the em into points might give a wrong result, since the font probably isn't set up yet, so that these dimensions haven't been updated, either.

```

1320 \def\MT@get@unit#1{%
1321 \expandafter\MT@get@unit@#1 e!\@nil
1322 \ifx\x\@empty\else\let#1\x\fi
1323 \@defaultunits\@tempdima#1 pt\relax\@nnil
1324 \ifdim\@tempdima=\z@
1325 \MT@warning@n1{%
1326 Cannot set \@nameuse{MT@abbr@MT@feat} factors relative to zero\MessageBreak
1327 width. Setting factors of list \@nameuse{MT@MT@feat @c@name}'\MessageBreak
1328 relative to character widths instead}%
1329 \let#1\@empty
1330 \let\MT@get@char@unit\MT@get@charwd
1331 \else
1332 \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} factors relative
1333 to \the\@tempdima}%
1334 \MT@count=\@tempdima\relax
1335 \fi
1336 }
1337 \def\MT@get@unit@#1e#2#3\@nil{%
1338 \ifx\#3\\\let\x\@empty \else
1339 \if m#2%
1340 \edef\x{#1\fontdimen6\MT@font}%
1341 \else
1342 \if x#2%
1343 \edef\x{#1\fontdimen5\MT@font}%
1344 \fi
1345 \fi
1346 \fi
1347 }

```

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

```

1348 \def\MT@set@inputenc#1{%

```

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

```

1349 \def\MT@cat{#1}%
1350 \edef\@tempa{MT@MT@feat @#1@\csname MT@MT@feat @#1@name\endcsname @inputenc}%
1351 \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1352 }

```

\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.

```

1353 \MT@addto@setup{%
1354 \@ifpackageloaded{inputenc}{%

```



```

1355 \ifpackageafter{inputenc}{2006/02/22}{%
1356 \def\MT@set@inputenc{%
1357 \MT@ifstreq\inputencodingname{\csname\@tempa\endcsname}\relax
1358 \MT@load@inputenc
1359 }%
1360 }%
1361 \let\MT@set@inputenc\MT@load@inputenc
1362 }%
1363 }%
1364 \def\MT@set@inputenc{%
1365 \MT@warning@n1{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1366 \MessageBreak package isn't loaded. Ignoring input encoding}%
1367 }%
1368 }%
1369 }

```

`\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.

```

1370 \def\MT@load@inputenc{%
1371 \MT@cfg@catcodes
1372 <debug>\MT@info@n1{1}{loading input encoding: \@nameuse{\@tempa}}%
1373 \inputencoding{\@nameuse{\@tempa}}%
1374 }
1375 </package>

```

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

```

1376 <*pdfTeX-def|xetex-def|luatex-def>
1377 \def\MT@set@pr@heirs#1{%
1378 \lcode\MT@font #1 =\lcode\MT@font\MT@char\relax
1379 \rcode\MT@font #1 =\rcode\MT@font\MT@char\relax
1380 <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
1381 <debug>\MT@info@n1{4}{;;; lp/rp (#1): \number\lcode\MT@font\MT@char\space/%
1382 <debug> \number\rcode\MT@font\MT@char\space}%
1383 }

```

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

```

\MT@preset@pr@ 1384 \def\MT@preset@pr{%
1385 \expandafter\expandafter\expandafter\MT@preset@pr@
1386 \csname \MT@pr@c@\MT@pr@c@name @preset\endcsname\@nil
1387 }
1388 \def\MT@preset@pr@#1,#2\@nil{%
1389 \ifx\MT@pr@unit@empty
1390 \MT@warn@preset@twidth{pr}%
1391 \let\MT@preset@aux\MT@preset@aux@factor
1392 \else
1393 \def\MT@preset@aux{\MT@preset@aux@space2}%
1394 \fi
1395 \MT@ifempty{#1}{\let\@tempa\empty}{\MT@preset@aux{#1}\@tempa}%
1396 \MT@ifempty{#2}{\let\@tempb\empty}{\MT@preset@aux{#2}\@tempb}%
1397 \MT@set@all@pr\@tempa\@tempb
1398 }

```

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

```

\MT@preset@aux@factor 1399 \def\MT@preset@aux@factor#1#2{%
\MT@preset@aux@space 1400 \@tempcntb=#1\relax
1401 \MT@scale@factor
1402 \edef#2{\number\@tempcntb}%
1403 }
1404 \def\MT@preset@aux@space#1#2#3{%
1405 \def\@tempb{#2}%
1406 \MT@get@space@unit#1%
1407 \MT@scale@to@em
1408 \edef#3{\number\@tempcntb}%
1409 }

```

`\MT@warn@preset@twidth`

```

1410 \def\MT@warn@preset@twidth#1{%
1411   \MT@warning@n1{%
1412     Cannot preset characters relative to their widths\MessageBreak
1413     for \@nameuse{MT@abbr#1} list ` \@nameuse{MT@#1@c@name}'. Presetting them%
1414     \MessageBreak relative to lem instead}%
1415   }
1416 \<pdfTeX-def|xetex-def|luatex-def>

```

14.2.2 Expansion

`\MT@expansion` Set up for expansion?

```

1417 \<*pdfTeX-def|luatex-def>
1418 \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).

```

1419 \def\MT@set@ex@codes@{%
1420   \MT@if@list@exists{%
1421     \MT@get@ex@opt
1422     \let\MT@get@char@unit\relax
1423     \MT@reset@ef@codes
1424     \MT@get@inh@list
1425     \MT@set@inputenc{c}%
1426     \MT@load@list\MT@ex@c@name
1427     \MT@set@listname
1428     \MT@let@cn\@tempc{MT@ex@c@\MT@ex@c@name}%
1429     \expandafter\MT@set@codes\@tempc,\relax,%
1430     \MT@expandfont
1431   }\relax
1432 }

```

`\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.

```

1433 \<pdfTeX-def|luatex-def>
1434 \<package>\newif\ifMT@nonselected
1435 \<*pdfTeX-def|luatex-def>
1436 \def\MT@set@ex@codes@n{%
1437   \MT@nonselectedtrue
1438   \MT@if@list@exists
1439   \MT@get@ex@opt
1440   {%
1441     \let\MT@stretch@ \MT@stretch
1442     \let\MT@shrink@ \MT@shrink
1443     \let\MT@step@ \MT@step
1444     \let\MT@auto@ \MT@auto
1445     \let\MT@ex@factor@ \MT@ex@factor
1446   }%
1447   \MT@reset@ef@codes
1448   \MT@expandfont
1449   \MT@nonselectedfalse
1450 }

```

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

```

1451 \let\MT@set@ex@codes\MT@set@ex@codes@n

```

`\MT@expandfont` Expand the font.

```
1452 \def\MT@expandfont{%
1453   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
1454 }
```

`\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).

```
1455 \def\MT@set@all@ex#1{%
1456   <debug>\MT@info@n1{3}{-- ex: setting all to \number#1}%
1457   \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
1458 }
1459 \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.

```
1460 <*pdfTeX-def>
1461 \MT@requires@pdfTeX4{
1462   \def\MT@reset@ef@codes{%
1463     \ifnum\MT@ex@factor@=\@m \else
1464       \MT@reset@ef@codes@
1465     \fi
1466   }
1467 }{
1468 </pdfTeX-def>
1469   \let\MT@reset@ef@codes\MT@reset@ef@codes@
1470 <pdfTeX-def>}
```

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

```
1471 \def\MT@ex@split@val#1\relax{%
1472   \@tempcntb=#1\relax
```

Take an optional factor into account.

```
1473   \ifnum\MT@ex@factor@=\@m \else
1474     \MT@scale\@tempcntb \MT@ex@factor@ \@m
1475   \fi
1476   \ifnum\@tempcntb > \MT@ex@max
1477     \MT@warn@ex@too@large\MT@ex@max
1478   \else
1479     \ifnum\@tempcntb < \MT@ex@min
1480       \MT@warn@ex@too@large\MT@ex@min
1481     \fi
1482   \fi
1483   \efcode\MT@font\MT@char=\@tempcntb
1484 <debug>\MT@info@n1{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%
```

Heirs, heirs, I love thy heirs.

```
1485 \MT@ifdefined@c@T\MT@ex@inh@name{%
1486   \MT@ifdefined@n@T{\MT@inh@\MT@ex@inh@name @\MT@char @}{%
1487     \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
1488   }%
1489 }%
1490 }
```

`\MT@warn@ex@too@large`

```
1491 \def\MT@warn@ex@too@large#1{%
1492   \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for
1493     character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
1494     Setting it to the maximum of \number#1}%
1495   \@tempcntb=#1\relax
1496 }
```

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

```
\MT@ex@factor@ 1497 \def\MT@get@ex@opt{%
\MT@stretch@ 1498   \MT@set@listname
\MT@shrink@
\MT@step@
\MT@auto@
```

```

1499 \MT@ifdefined@n@TF{MT@ex@c@MT@ex@c@name @factor}{%
1500 \MT@let@cn\MT@ex@factor@{MT@ex@c@MT@ex@c@name @factor}%
1501 \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor@/1000}%
1502 }{%
1503 \let\MT@ex@factor@MT@ex@factor
1504 }%
1505 \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
1506 \MT@get@ex@opt@{shrink}{Setting shrink limit to \number\MT@shrink@}%
1507 \MT@get@ex@opt@{step}{Setting expansion step to \number\MT@step@}%
1508 \def@tempa{autoexpand}%
1509 \MT@get@ex@opt@{auto}{\ifx\@tempa\MT@auto@ En\else Dis\fi abling automatic expansion}%
1510 \MT@ifdefined@n@T{MT@ex@c@MT@ex@c@name @preset}{%
1511 \MT@preset@ex
1512 \let\MT@reset@ef@codes\relax
1513 }%
1514 }

```

\MT@get@ex@opt@

```

1515 \def\MT@get@ex@opt@#1#2{%
1516 \MT@ifdefined@n@TF{MT@ex@c@MT@ex@c@name @#1}{%
1517 \MT@let@nn{MT@#1@}{MT@ex@c@MT@ex@c@name @#1}%
1518 \MT@vinfo{... : #2}%
1519 }{%
1520 \MT@let@nn{MT@#1@}{MT@#1}%
1521 }%
1522 }

```

\MT@set@ex@heirs

```

1523 \def\MT@set@ex@heirs#1{%
1524 \efcode\MT@font#1=\efcode\MT@font\MT@char
1525 <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
1526 <debug>\MT@info@n1{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
1527 }

```

\MT@preset@ex

```

1528 \def\MT@preset@ex{%
1529 \@tempcntb=\csname MT@ex@c@MT@ex@c@name @preset\endcsname\relax
1530 \MT@scale@factor
1531 \MT@set@all@ex@\@tempcntb
1532 }
1533 </pdfTeX-def|luatex-def>

```

14.2.3 Interword spacing (glue)

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

```

1534 <*pdfTeX-def>
1535 \MT@requires@pdfTeX6{
1536 \def\MT@spacing{\MT@maybe@do{sp}}

```

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

```

1537 \def\MT@set@sp@codes{%
1538 \MT@if@list@exists{%
1539 \MT@get@font@dimen@six{%
1540 \MT@get@opt
1541 \MT@reset@sp@codes
1542 \MT@get@inh@list
1543 \MT@set@inputenc{c}%
1544 \MT@load@list\MT@sp@c@name
1545 \MT@set@listname
1546 \MT@let@cn\@tempc{MT@sp@c@MT@sp@c@name}%
1547 \expandafter\MT@set@codes\@tempc,\relax,}%
1548 }\MT@reset@sp@codes
1549 }

```

`\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).

```

1550 \def\MT@sp@split@val#1,#2,#3\relax{%
1551   \def\@tempb{#1}%
1552   \MT@ifempty\@tempb\relax{%
1553     \MT@get@space@unit2%
1554     \MT@scale@to@em
1555     \knbscode\MT@font\MT@char=\@tempcntb
1556     <debug>\MT@info{n}{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
1557   }%
1558   \def\@tempb{#2}%
1559   \MT@ifempty\@tempb\relax{%
1560     \MT@get@space@unit3%
1561     \MT@scale@to@em
1562     \stbscode\MT@font\MT@char=\@tempcntb
1563     <debug>\MT@info{n}{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
1564   }%
1565   \def\@tempb{#3}%
1566   \MT@ifempty\@tempb\relax{%
1567     \MT@get@space@unit4%
1568     \MT@scale@to@em
1569     \shbscode\MT@font\MT@char=\@tempcntb
1570     <debug>\MT@info{n}{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
1571   }%
1572   \MT@ifdefined@c@T\MT@sp@inh@name{%
1573     \MT@ifdefined@nT\MT@inh@\MT@sp@inh@name @\MT@char @}%
1574     \MT@exp@cs\MT@map@tlist@c\MT@inh@\MT@sp@inh@name @\MT@char @\MT@set@sp@heirs
1575   }%
1576 }%
1577 }

```

`\MT@set@sp@heirs`

```

1578 \def\MT@set@sp@heirs#1{%
1579   \knbscode\MT@font#1=\knbscode\MT@font\MT@char
1580   \stbscode\MT@font#1=\stbscode\MT@font\MT@char
1581   \shbscode\MT@font#1=\shbscode\MT@font\MT@char
1582   <debug>\MT@info{n}{2}{-- heir of \MT@char: #1}%
1583   <debug>\MT@info{n}{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
1584   <debug> \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
1585 }

```

`\MT@set@all@sp`

```

\MT@reset@sp@codes 1586 \def\MT@set@all@sp#1#2#3{%
1587   <debug>\MT@info{n}{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
\MT@reset@sp@codes@ 1588   \let\MT@temp\@empty
1589   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
1590   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
1591   \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
1592   \MT@do@font\MT@temp
1593 }
1594 \def\MT@reset@sp@codes@{\MT@set@all@sp\z@\z@\z@}
1595 \let\MT@reset@sp@codes\relax

```

`\MT@preset@sp`

```

\MT@preset@sp@ 1596 \def\MT@preset@sp{%
1597   \expandafter\expandafter\expandafter\MT@preset@sp@
1598   \csname MT@sp@c@\MT@sp@c@name @preset\endcsname\@nil
1599 }
1600 \def\MT@preset@sp@#1,#2,#3\@nil{%
1601   \ifx\MT@sp@unit@\@empty
1602     \MT@warn@preset@to@width{sp}%
1603     \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
1604     \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
1605     \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%

```

```

1606 \else
1607 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
1608 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
1609 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
1610 \fi
1611 \MT@set@all@sp\@tempa\@tempc\@tempb
1612 }
1613 }\relax

```

14.2.4 Additional kerning

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

```

1614 \MT@requires@pdftex6{
1615 \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

1616 \def\MT@set@kn@codes{%
1617 \MT@if@list@exists{%
1618 \MT@get@font@dimen@six{%
1619 \MT@get@opt
1620 \MT@reset@kn@codes
1621 \MT@get@inh@list
1622 \MT@set@inputenc{c}%
1623 \MT@load@list\MT@kn@c@name
1624 \MT@set@listname
1625 \MT@let@cn\@tempc{MT@kn@c@\MT@kn@c@name}%
1626 \expandafter\MT@set@codes\@tempc,\relax,%
1627 }\MT@reset@kn@codes
1628 }

```

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

```

1629 \def\MT@kn@split@val#1,#2\relax{%
1630 \def\@tempb{#1}%
1631 \MT@ifempty\@tempb\relax{%
1632 \MT@get@space@unit2%
1633 \MT@scale@to@em
1634 \knbcode\MT@font\MT@char=\@tempcntb
1635 <debug>\MT@dinfoln{4}{;;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
1636 }%
1637 \def\@tempb{#2}%
1638 \MT@ifempty\@tempb\relax{%
1639 \MT@get@space@unit2%
1640 \MT@scale@to@em
1641 \knacode\MT@font\MT@char=\@tempcntb
1642 <debug>\MT@dinfoln{4}{;;; knac (\MT@char): \number\knacode\MT@font\MT@char: [#2]}%
1643 }%
1644 \MT@ifdefined@c@T\MT@kn@inh@name{%
1645 \MT@ifdefined@nT{MT@inh@\MT@kn@inh@name @\MT@char @}{%
1646 \MT@exp@cs\MT@map@tlist@c{MT@inh@\MT@kn@inh@name @\MT@char @}\MT@set@kn@heirs
1647 }%
1648 }%
1649 }

```

\MT@set@kn@heirs

```

1650 \def\MT@set@kn@heirs#1{%
1651 \knbcode\MT@font#1=\knbcode\MT@font\MT@char
1652 \knacode\MT@font#1=\knacode\MT@font\MT@char
1653 <debug>\MT@dinfoln{2}{-- heir of \MT@char: #1}%
1654 <debug>\MT@dinfoln{4}{;;; knbc (#1): \number\knbcode\MT@font\MT@char/%
1655 <debug> \number\knacode\MT@font\MT@char}%
1656 }

```

\MT@set@all@kn

\MT@reset@kn@codes

\MT@reset@kn@codes@

```

1657 \def\MT@set@all@kn#1#2{%
1658 <debug>\MT@info{n}{3}{-- knac/knbc: setting all to #1/#2}%
1659 \let\MT@temp\@empty
1660 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbcode\MT@font\@tempcnta=#1\relax}}%
1661 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knaccode\MT@font\@tempcnta=#2\relax}}%
1662 \MT@do@font\MT@temp
1663 }
1664 \def\MT@reset@kn@codes@{\MT@set@all@kn\z@\z@}
1665 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn@ 1666 \def\MT@preset@kn{%
1667 \expandafter\expandafter\expandafter\MT@preset@kn@
1668 \csname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
1669 }
1670 \def\MT@preset@kn@#1,#2\@nil{%
1671 \ifx\MT@kn@unit@\@empty
1672 \MT@warn@preset@tewidth{kn}%
1673 \let\MT@preset@aux\MT@preset@aux@factor
1674 \else
1675 \def\MT@preset@aux{\MT@preset@aux@space2}%
1676 \fi
1677 \MT@ifempty{#1}\let\@tempa\@empty{\MT@preset@aux{#1}\@tempa}%
1678 \MT@ifempty{#2}\let\@tempb\@empty{\MT@preset@aux{#2}\@tempb}%
1679 \MT@set@all@kn\@tempa\@tempb
1680 }
1681 }\relax
1682 </pdfTeX-def>

```

14.2.5 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```

1683 <*pdfTeX-def|luatex-def>
1684 <pdfTeX-def>\MT@requires@pdfTeX6
1685 <luatex-def>\MT@requires@luatex3
1686 {

```

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

```

\MT@tr@font@list 1687 \let\MT@tr@font@list\@empty
1688 \def\MT@tracking@{%
1689 \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
1690 \ifMT@inlist\@else
1691 \MT@maybe@do{tr}%
1692 \ifMT@do\else
1693 \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
1694 \fi
1695 \fi
1696 }
1697 </pdfTeX-def|luatex-def>
1698 <pdfTeX-def|luatex-def|letterspace>\let\MT@tracking
1699 <pdfTeX-def|luatex-def> \MT@tracking@
1700 <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.

```

1701 <*pdfTeX-def|luatex-def|letterspace>
1702 \def\MT@set@tr@codes{%
1703 <*pdfTeX-def|luatex-def>
1704 \MT@vinfo{Tracking font `'\MT@font'\on@line}%
1705 \MT@get@font@dimen@six{%
1706 \MT@if@list@exists
1707 \MT@get@tr@opt

```

```

1708 \relax
1709 </pdfTeX-def| LaTeX-def>
1710 \MT@ifdefined@c@TF\MT@letterspace@relax{\let\MT@letterspace@MT@letterspace}%
1711 \ifnum\MT@letterspace@=z@

```

Zero tracking requires special treatment.

```

1712 \MT@set@tr@zero
1713 \else
1714 <pdfTeX-def| LaTeX-def> \MT@vinfo{... Tracking by \number\MT@letterspace}%

```

Letterspacing only works in PDF mode.

```

1715 \MT@warn@tracking@DVI

```

\MT@lsfont The letterspaced font instances are saved in macros \/<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.

```

1716 \xdef\MT@lsfont{\csname\expandafter\string\font@name
1717 \number\MT@letterspace@ls\endcsname}%
1718 \expandafter\ifx\MT@lsfont\relax
1719 <debug>\MT@info@nl{1}{... new letterspacing instance}%

```

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

```

1720 \MT@get@ls@basefont
1721 \global\expandafter\letterspacefont\MT@lsfont\font@name\MT@letterspace@

```

Scale interword spacing (not configurable in letterspace).

```

1722 <*pdfTeX-def| LaTeX-def>
1723 \MT@ifdefined@c@TF\MT@tr@ispace
1724 {\let\@tempa\MT@tr@ispace}%
1725 {\edef\@tempa{\MT@letterspace@*,,}}%
1726 \MT@ifdefined@c@TF\MT@tr@ospace
1727 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
1728 {\edef\@tempa{\@tempa,,,}}%
1729 \expandafter\MT@tr@set@space\@tempa,%
1730 </pdfTeX-def| LaTeX-def>
1731 <*letterspace>
1732 % spacing = {<letterspace amount>*,,}
1733 \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@letterspace@\relax sp
1734 * \fontdimen2\MT@lsfont/1000\relax
1735 </letterspace>

```

Adjust outer kerning (microtype only).

```

1736 <*pdfTeX-def| LaTeX-def>
1737 \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,,}}%
1738 \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

1739 \MT@ifdefined@c@TF\MT@tr@ligatures\MT@tr@noligatures
1740 </pdfTeX-def| LaTeX-def>
1741 <*letterspace>
1742 % no ligatures = {f}
1743 \tagcode\MT@lsfont`f=\m@ne
1744 </letterspace>

```

Adjust protrusion values now, and maybe later (in \MT@pr@split@val).

```

1745 <debug>\MT@info@nl{2}{... compensating for tracking (\number\MT@letterspace@)}%
1746 \MT@do@font{\lcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax
1747 \rprcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax}%
1748 <pdfTeX-def| LaTeX-def> \let\MT@the@pr@code\MT@the@pr@code@tr
1749 \fi

```


Finally, let the letterspaced font propagate.

```
1750 \aftergroup\MT@set@lsfont
1751 <pdfTeX-def|luatex-def> \let\MT@font\MT@lsfont
```

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

```
\MT@curr@ls 1752 \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@letterspace@}}%
1753 \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.

```
1754 <pdfTeX-def|luatex-def>
1755 \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
1756 \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
1757 \MT@tr@outer@l
1758 </pdfTeX-def|luatex-def>
```

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

```
1759 \ifx\MT@ls@adjust@empty
1760 <letterspace> % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
1761 \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\font@name/2000\relax
1762 \MT@ls@outer@k
1763 <letterspace>
1764 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
1765 \aftergroup\aftergroup\aftergroup\MT@ls@aftergroup
1766 </letterspace>
```

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

```
1767 <pdfTeX-def|luatex-def>
1768 \else
1769 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
1770 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1771 \ifdim\MT@outer@kern=z@\else \MT@ls@outer@k \fi
1772 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
1773 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1774 </pdfTeX-def|luatex-def>
1775 \fi
1776 <pdfTeX-def|luatex-def>
```

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

```
1777 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
1778 \aftergroup\aftergroup\aftergroup\MT@ls@aftergroup
1779 </pdfTeX-def|luatex-def>
1780 \fi
1781 <pdfTeX-def|luatex-def> }%
1782 }
```

\MT@ls@aftergroup Stuff to be done after the letterspace group. The letterspace package only adjusts the kerning.

```
1783 <letterspace>\def\MT@ls@aftergroup{\MT@set@curr@ok\MT@ls@outer@k}
```

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

```
1784 </pdfTeX-def|luatex-def|letterspace>
1785 <package>
1786 \MT@addto@setup{%
1787 \ifpackageloaded{tikz}
1788 {\def\MT@ls@aftergroup{%
```

```

1789     \ifnum\tikz@expandcount>\z@ \else
1790     \MT@set@curr@os\MT@set@curr@ok\expandafter\MT@tr@outer@r\fi}}
1791     {\def\MT@ls@aftergroup{\MT@set@curr@os\MT@set@curr@ok\MT@tr@outer@r}}}}
1792 \end{package}
1793 \ifx\pdftex-def\luatex-def
\MT@get@tr@opt    Various settings (only for the microtype version).
1794 \def\MT@get@tr@opt{%
1795     \MT@set@listname
1796     \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name}{%
1797         \MT@let@cn\MT@letterspace{MT@tr@c@\MT@tr@c@name}%
\MT@tr@unit@    Different unit?
1798     \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name @unit}{%
1799         \MT@let@cn\MT@tr@unit@{MT@tr@c@\MT@tr@c@name @unit}%
1800         \ifdim\MT@tr@unit@=1em
1801             \let\MT@tr@unit@=\undefined
1802         \else
1803             \MT@let@cn\@tempb{MT@tr@c@\MT@tr@c@name}%
1804             \MT@get@unit\MT@tr@unit@
1805             \let\MT@tr@factor@\@m
1806             \MT@scale@to@em
1807             \edef\MT@letterspace{\number\@tempcntb}%
1808             \fi
1809         }%
1810     }%
\MT@tr@ispace    Adjust interword spacing.
\MT@tr@ospace 1811 \MT@get@tr@opt@{spacing} {ispace}%
1812 \MT@get@tr@opt@{outerspacing}{ospace}%
\MT@tr@okern    Adjust outer kerning.
1813 \MT@get@tr@opt@{outerkerning}{okern}%
\MT@tr@ligatures    Which ligatures should we disable (empty means all, undefined none)?
1814 \MT@get@tr@opt@{noligatures} {ligatures}%
1815 }
\MT@get@tr@opt@
1816 \def\MT@get@tr@opt@#1#2{%
1817     \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name @#1}%
1818     {\MT@let@nn{MT@tr@c@#2}{MT@tr@c@\MT@tr@c@name @#1}}%
1819 }
1820 \ifx\pdftex-def\luatex-def
\MT@set@lsfont    Redefine \font@name, which will be called a second later (in \selectfont).
1821 \ifx\pdftex-def\luatex-def\letterspace
1822 \ifx\plain\MT@requires@latex2{
1823 \def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}
\lsstyle    Disable the tests whether the font should be letterspaced, then trigger the setup.
Only \textls can be used in math mode (\lsstyle may be used inside another
text switch, of course). Still, we have to ensure that math fonts are set up again.
1824 \DeclareRobustCommand\lsstyle{%
1825     \not@math@alphabet\lsstyle\textls
1826     \let\glb@currsizel@empty
1827     \ifx\pdftex-def\luatex-def \def\MT@feat{tr}%
1828     \let\MT@tracking\MT@set@tr@codes
1829     \selectfont
1830 }
Now the definitions for the letterspace package with plain TEX.
1831 \ifx\plain
1832 }{

```

```

1833 \def\MT@set@lsfont{\MT@lsfont}
1834 \def\lsstyle{%
1835   \begingroup
1836   \escapechar\m@ne
1837   \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
1838   \MT@set@tr@codes
1839   \endgroup
1840 }
1841 \let\textls\undefined
1842 \let\lslig\undefined
1843 }
1844 </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.

```

1845 \DeclareRobustCommand\lslig[1]{%
1846   {\MT@ifdefined@c@TF\MT@curr@ls{%
1847     \escapechar\m@ne
1848     \MT@get@ls@basefont
1849     \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax
1850     \kern\MT@outer@kern
1851     \font@name #1%
1852     \kern\MT@outer@kern%
1853   }{#1}}}%
1854 }

```

`\MT@ls@basefont` pdfTeX 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.

```

1855 \def\MT@get@ls@basefont{%
1856   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1857   \expandafter\ifx\MT@ls@basefont\relax
1858     \MT@exp@two@c\MT@gl@et\MT@ls@basefont\font@name
1859   \else
1860     <debug>\MT@din@fo@n{1}{... fixing base font}%
1861     \MT@exp@two@c\let\font@name\MT@ls@basefont
1862   \fi
1863 }

```

`\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`

```

1864 \def\MT@set@ls@basefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}
1865 \def\MT@set@tr@zero{%
1866   <debug>\MT@din@fo@n{1}{... zero tracking}%
1867   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1868   \expandafter\ifx\MT@ls@basefont\relax \else
1869     <debug>\MT@din@fo@n{1}{... fixing base font}%
1870     \aftergroup\MT@set@ls@basefont
1871   \fi
1872 }

```

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

```

1873 </pdfTeX-def|luatex-def|letterspace>
1874 <*pdfTeX-def|luatex-def>
1875 <pdfTeX-def>\MT@requires@pdfTeX7{
1876   \def\MT@tr@noligatures{%
1877     \ifx\MT@tr@ligatures\@empty
1878       \MT@noligatures@\MT@lsfont\undefined
1879     \else
1880       \MT@noligatures@\MT@lsfont\MT@tr@ligatures
1881     \fi

```

```

1882 }
1883 <*/pdfTeX-def>
1884 }{
1885   \def\MT@tr@noligatures{%
1886     \MT@warning@n1{%
1887       Disabling selected ligatures is only possible since\MessageBreak
1888       pdfTeX 1.40.4. Disabling all ligatures instead}%
1889     \MT@gl@t\MT@tr@noligatures\relax
1890   }
1891 }
1892 </pdfTeX-def>

```

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

```
1893 \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.

```

1894 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
1895   <debug>\MT@din@n12{... orig. space: \the\fontdimen2\MT@lsfont,
1896   <debug>   \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
1897   <debug>   \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
1898   \let\MT@temp\empty
1899   \MT@tr@set@space@{#1}{#4}{2}\empty
1900   \MT@tr@set@space@{#2}{#5}{3}\@plus
1901   \MT@tr@set@space@{#3}{#6}{4}\@minus
1902   \MT@gl@t@nc{\MT@outer@space\expandafter\string\font@name}\MT@temp
1903   <debug>\MT@din@n12{... inner space: \the\fontdimen2\MT@lsfont,
1904   <debug>   \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
1905   <debug>\MT@din@n12{... outer space: \MT@temp}%
1906 }

```

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

```

1907 \def\MT@tr@set@space@#1#2#3#4{%
1908   \MT@ifempty{#2}{%
1909     \MT@ifempty{#1}{%
1910       \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
1911     }%
1912     \MT@tr@set@space@@{#1}{#3}{1000}%
1913     \edef\MT@temp{\MT@temp#4\the\@tempdima}%
1914     \fontdimen#3\MT@lsfont=\@tempdima
1915   }%
1916 }%
1917 \MT@tr@set@space@@{#2}{#3}{2000}%
1918 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
1919 \MT@ifempty{#1}\relax{%
1920   \MT@tr@set@space@@{#1}{#3}{1000}%
1921   \fontdimen#3\MT@lsfont=\@tempdima
1922 }%
1923 }%
1924 }

```

\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.

```

1925 \def\MT@tr@set@space@@#1#2#3{%
1926   \MT@test@ast#1*\@nil{%
1927     \MT@ifdefined@c@TF\MT@tr@unit@
1928     {\edef\@tempb{#1}\MT@scale@to@em}
1929     {\@tempcntb=#1\relax}%
1930     \@tempdima=\dimexpr \dimexpr\@tempcntb sp*\MT@dimen@six/1000\relax
1931     -\fontdimen#2\MT@lsfont\relax

```

For \fontdimen 2, we also have to subtract the kerning that letterspacing adds to

the sides of the characters (only half if it's for outer spacing).

```

1932 \ifnum#2=\tw@
1933 \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
1934 \fi
1935 \@tempdima=\dimexpr \fontdimen#2\MT@lsfont+\@tempdima\relax
1936 }{%
1937 \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
1938 \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
1939 }%
1940 <debug>\MT@info{n13{... : font dimen #2 (#1): \the\@tempdima}%
1941 }
```

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

```

1942 \def\MT@tr@outer@l{%
1943 \ifhmode
1944 \ifdim\lastskip>5sp
1945 \edef\x{\the\lastskip minus 0pt}%
1946 \setbox\z@\hbox{\MT@outer@space=\x}%
1947 \ifdim\wd\z@>\z@
1948 <debug>\MT@info2{[[[ adjusting pre space: \the\MT@outer@space}%
1949 \unskip \hskip\MT@outer@space\relax
```

Disable left outer kerning.

```

1950 \let\MT@ls@outer@k\relax
1951 \else
```

The ragged2e package sets `\spaceskip` without glue.

```

1952 \ifdim\lastskip=%
1953 \ifnum\spacefactor<2000
1954 \spaceskip
1955 \else
1956 \ifdim\xspaceskip=\z@
1957 \dimexpr\spaceskip+\fontdimen7\font@name\relax
1958 \else
1959 \xspaceskip
1960 \fi
1961 \fi
1962 <debug>\MT@info2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
1963 \unskip \hskip\MT@outer@space\relax
1964 \let\MT@ls@outer@k\relax
1965 \fi
1966 \fi
1967 \fi
1968 \fi
1969 }
```

`\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`).

`\MT@tr@outer@r`

```

1970 \def\MT@tr@outer@r{%
1971 \futurelet\MT@tr@outer@next\MT@tr@outer@r@
1972 }
```

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

```

1973 \def\MT@if@outer@next#1{%
1974 \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
1975 }
```

`\MT@tr@outer@r@`

```

1976 \def\MT@tr@outer@r@{%
1977 \def\MT@temp*{}%
```

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

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).

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.

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).

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

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

For older pdf_{TEX} versions and Lua_{TEX}, throw an error.

2021 } {

```

2022 \DeclareRobustCommand\lsstyle{%
2023   \MT@error{Letterspacing only works with \MT@engine tex version
2024   \pdfTeX-def 1.40%
2025   \luatex-def 0.62%
2026   \MessageBreak or newer}
2027   {Upgrade \MT@engine tex, or try the `soul' package instead.}%
2028   \MT@glet\lsstyle\relax
2029 }
2030 }

```

And for Xe_{La}TeX, too.

```

2031 \pdfTeX-def \luatex-def
2032 \xetex-def
2033 \DeclareRobustCommand\lsstyle{%
2034   \MT@error{Letterspacing currently doesn't work with xetex}
2035   {Run pdftex or luatex, or use the `soul' package instead.}%
2036   \MT@glet\lsstyle\relax
2037 }
2038 \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.

```

2039 \package letterspace
2040 \DeclareRobustCommand\textls{%
2041   \ifstar{\let\MT@ls@adjust@MT@ls@adjust@empty\MT@textls}%
2042   {\let\MT@ls@adjust@MT@ls@adjust@relax\MT@textls}%
2043 }

```

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

```

2044 \newcommand\MT@textls[2][{}]{%
2045   \ifmmode
2046     \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
2047   \else
2048     \hmode@bgroup
2049     \MT@ls@set@ls{#1}%
2050     \lsstyle #2%
2051     \expandafter
2052     \egroup
2053   \fi
2054 }

```

`\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
2055 \def\MT@ls@adjust@empty{\let\MT@ls@adjust\@empty}
2056 \def\MT@ls@adjust@relax{\let\MT@ls@adjust\relax}
2057 \def\MT@ls@set@ls#1{%
2058   \MT@ifempty{#1}%
2059   {\let\MT@letterspace@ \@undefined}%
2060   {\KV@sp@def\MT@letterspace@{#1}%
2061     \edef\MT@letterspace@{\number\MT@letterspace@}%
2062     \MT@ls@too@large\MT@letterspace@}%
2063   \MT@ls@adjust@
2064 }

```

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

```

2065 \def\MT@ls@too@large#1{%
2066   \ifnum#1>\MT@tr@max
2067     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
2068     \let#1\MT@tr@max
2069   \else
2070     \ifnum#1<\MT@tr@min

```

```

2071      \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
2072      \let#1\MT@tr@min
2073      \fi
2074      \fi
2075  }

\MT@outer@kern      This dimen is used for the starred version of \textls, for \lslig and for adjusted
\MT@tr@set@okern    outer kerning.
2076 \newdimen\MT@outer@kern
2077 </package|letterspace>
2078 <*pdfTeX-def|luatex-def>
2079 \def\MT@tr@set@okern#1,#2,{%
2080   \let\MT@temp@empty
2081   \MT@ifempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
2082   \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
2083   \MT@gl@et@nc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
2084   <debug>\MT@info@n12{... outer kerning: (#1,#2)
2085   <debug>           = \nameuse{\MT@outer@kern\expandafter\string\font@name}}%
2086 }

\MT@tr@set@okern@
2087 \def\MT@tr@set@okern@#1{%
2088   \MT@test@ast#1*\@nil{%
2089     \MT@ifdefined@c@TF\MT@tr@unit@
2090     {\edef\@tempb{#1}\MT@scale@to@em}
2091     {\@tempcntb=#1\relax}%
2092     \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
2093   }{%
2094     \MT@ifempty\@tempa{\let\@tempa\@m}\relax
2095     \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
2096     * \fontdimen6\MT@lsfont/2000\relax
2097   }%
2098   \advance\@tempdima -\dimexpr \MT@letterspace@ sp
2099   * \fontdimen6\MT@lsfont/2000\relax
2100   \edef\MT@temp{\MT@temp{\the\@tempdima}}%
2101 }
2102 </pdfTeX-def|luatex-def>

\MT@ls@outer@k      Adjust outer kerning.
2103 <pdfTeX-def|luatex-def|letterspace>\def\MT@ls@outer@k{\ifhmode\kern\MT@outer@kern\relax\fi}
2104 <*pdfTeX-def|luatex-def>

```

14.2.6 Disabling ligatures

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

```

2105 <pdfTeX-def>\MT@requires@pdfTeX5{
2106 \def\MT@noligatures{%
2107   \MT@dotrue
2108   \let\@tempa\MT@n1@setname
2109   \MT@map@c@list@n{font,encoding,family,series,shape,size}{%
2110     \MT@ifdefined@c@TF{\MT@checklist@##1}%
2111     {\csname MT@checklist@##1\endcsname}%
2112     {\MT@checklist@{##1}}}%
2113   {nl}}%
2114 }%
2115 \ifMT@do
2116   \MT@noligatures@{\MT@font\MT@n1@ligatures
2117   \fi
2118 }

\MT@noligatures@    This is also used by \MT@set@tr@codes.
2119 \def\MT@noligatures@#1#2{%
2120   \MT@ifdefined@c@TF#2{%

```


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

```

2121 \MT@ifdefined@ccTF\tagcode{%
    No 'inputenc' key.
2122 \let\MT@warn@maybe@inputenc\@empty
2123 \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
2124 \MT@map@clist@c#2{%
2125 \KV@sp@def\@tempa{##1}\MT@get@slot
2126 \ifnum\MT@char>\m@ne \tagcode#1\MT@char=\m@ne \fi}%
2127 \MT@vinfo{... Disabling ligatures for characters: #2}%
2128 }{%
2129 \pdfnoligatures#1%
2130 \MT@warning{Cannot disable selected ligatures (pdftex doesn't\MessageBreak
2131 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
2132 the font instead}%
2133 }%
2134 }{%
2135 \pdfnoligatures#1%
2136 \MT@vinfo{... Disabling ligatures}%
2137 }%
2138 }
2139 <pdfTeX-def>\relax
2140 <pdfTeX-def|luatex-def>

```

14.2.7 Loading the configuration

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

```

2141 <package>
2142 \def\MT@load@list#1{%
2143 \edef\@tempa{#1}%
2144 \MT@let@cn\@tempb{\MT@MT@feat @c@\@tempa @load}%
2145 \MT@ifstreq\@tempa\@tempb{%
2146 \MT@error{\@nameuse{\MT@abbr@\MT@feat} list \@tempa' cannot load itself}{}%
2147 }{%
2148 \ifx\@tempb\relax \else
2149 \MT@ifdefined@n@TF{\MT@\MT@feat @c@\@tempb}{%
2150 \MT@vinfo{... : First loading \@nameuse{\MT@abbr@\MT@feat} list \@tempb'}%
2151 \begingroup
2152 \MT@load@list\@tempb
2153 \endgroup
2154 \edef\MT@curr@list@name{\@nameuse{\MT@abbr@\MT@feat} list
2155 \noexpand\MessageBreak\@tempb'}%
2156 \MT@let@cn\@tempc{\MT@\MT@feat @c@\@tempb}%
2157 \expandafter\MT@set@codes\@tempc,\relax,%
2158 }{%
2159 \MT@error{\@nameuse{\MT@abbr@\MT@feat} list \@tempb' undefined.\MessageBreak
2160 Cannot load it from list \@tempa'}{}%
2161 }%
2162 \fi
2163 }%
2164 }

```

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

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

```

2165 \let\MT@file@list\@empty
2166 \def\MT@find@file#1{%
    Check for existence of the file only once.
2167 \MT@in@clist{#1}\MT@file@list
2168 \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.

```

2169 \MT@begin@catcodes
2170 \let\MT@begin@catcodes\relax
2171 \let\MT@end@catcodes\relax
2172 \InputIfFileExists{mt-#1.cfg}{%
2173 \edef\MT@curr@file{mt-#1.cfg}%
2174 \MT@vinfo{... Loading configuration file \MT@curr@file}%
2175 \MT@xadd\MT@file@list{#1,}%
2176 }{%
2177 \MT@get@basefamily#1\@empty\@empty\@empty\@nil
2178 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
2179 \ifMT@inlist@
2180 \MT@xadd\MT@file@list{#1,}%
2181 \else
2182 \InputIfFileExists{mt-\@tempa.cfg}{%
2183 \edef\MT@curr@file{mt-\@tempa.cfg}%
2184 \MT@vinfo{... Loading configuration file \MT@curr@file}%
2185 \MT@xadd\MT@file@list{\@tempa,#1,}%
2186 }{%
2187 \MT@vinfo{... No configuration file mt-#1.cfg}%
2188 \MT@xadd\MT@file@list{#1,}%
2189 }%
2190 \fi
2191 }%
2192 \endgroup
2193 \fi
2194 }

```

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

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

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

```

2195 \def\MT@cfg@catcodes{%
2196 \makeatletter
2197 \catcode\^7%
2198 \catcode\ 9%
2199 \catcode\^^I9%
2200 \catcode\^^M9%
2201 \catcode\\\z@
2202 \catcode\{\@ne
2203 \catcode\}\tw@
2204 \catcode\#6%
2205 \catcode\%14%
2206 \MT@map@tlist@n
2207 {\!\"$&'(\)\*+,\-\.\/\:\;\<=\>\?[\\\_-\|\~}%
2208 \makeother
2209 }

```

`\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.

```

2210 \def\MT@begin@catcodes{%
2211 \begingroup
2212 \MT@cfg@catcodes
2213 }

```

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

```

2214 \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)

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	•	-	•	-	•	-	•	-	•	-	•	-	•	-	•	-

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.

```

2215 \def\MT@get@basefamily#1#2#3#4\@nil{%
2216   \ifx\@empty#4%
2217     \def\@tempa{#1#2#3}%
2218   \else
2219     \let\@tempa\@empty
2220     \edef\@tempb{#1#2#3#4}%
2221     \expandafter\MT@get@basefamily@\@tempb\@nil
2222   \fi
2223 }
```

\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'.

```

2224 \def\MT@get@basefamily@#1#2\@nil{%
2225   \edef\@tempa{\@tempa#1}%
2226   \ifx\#2\expandafter\@gobble\else\expandafter\@firstofone\fi
2227   {\MT@in@tlist{#2}\MT@variants
2228    \ifMT@inlist\else\MT@get@basefamily@#2\@nil\fi}%
2229 }
```

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

```

\MT@get@listname
2230 \def\MT@get@listname#1{%
2231   <debug>\MT@info{n1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
2232   \let\MT@listname\@undefined
2233   \def\@tempb{#1}%
2234   \MT@map@tlist@c\MT@try@order\MT@get@listname@
2235 }
2236 \def\MT@get@listname@#1{%
2237   \expandafter\MT@next@listname#1%
2238   \ifx\MT@listname\@undefined \else
2239     \expandafter\MT@tlist@break
2240   \fi
2241 }
```

\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.

```

2242 \def\MT@try@order{%
2243   {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
2244   {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
2245 }
```

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

```

2246 \def\MT@next@listname#1#2#3#4{%
2247   \ifnum#1=\z@\MT@nofamilytrue\fi
2248   \edef\@tempa{\MT@encoding
2249   /\ifnum#1=\@ne \MT@family \fi
```

```

2250 /\ifnum#2=\@ne \MT@series \fi
2251 /\ifnum#3=\@ne \MT@shape \fi
2252 /\ifnum#4=\@ne *\fi
2253 \MT@context}%
2254 <debug>\MT@info{n}{1}{trying \@tempa}%
2255 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
2256 \MT@next@listname@#4%
2257 }{%

```

Also try with an alias family.

```

2258 \ifnum#1=\@ne
2259 \ifx\MT@familyalias\@empty \else
2260 \edef\@tempa{\MT@encoding
2261 \MT@familyalias
2262 /\ifnum#2=\@ne \MT@series\fi
2263 /\ifnum#3=\@ne \MT@shape\fi
2264 /\ifnum#4=\@ne *\fi
2265 \MT@context}%
2266 <debug>\MT@info{n}{1}{(alias) \@tempa}%
2267 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
2268 \MT@next@listname@#4%
2269 }%
2270 \fi
2271 \fi
2272 }%
2273 }

```

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

```

2274 \def\MT@next@listname@#1{%
2275 \ifnum#1=\@ne
2276 \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @sizes}%
2277 \ifMT@inlist@
2278 \let\MT@listname\MT@size@name
2279 \fi
2280 \else
2281 \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
2282 \fi
2283 }

```

\MT@if@list@exists

```

\MT@context 2284 \def\MT@if@list@exists{%
2285 \MT@let@cn\MT@context{MT@\MT@feat @context}%
2286 \MT@ifstreq{@}\MT@context{\let\MT@context\@empty}\relax
2287 \MT@get@listname{\MT@feat @c}%
2288 \MT@ifdefined@c@TF\MT@listname{%
2289 \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
2290 \ifMT@nonselected
2291 \MT@vinfo{... Applying non-selected expansion (list '\MT@listname')}%
2292 \else
2293 \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list '\MT@listname'}%
2294 \fi
2295 \@firstoftwo
2296 }{%

```

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

```

2297 \MT@let@cn{MT@\MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```

2298 \ifMT@nonselected
2299 \MT@vinfo{... Applying non-selected expansion (no list)}%
2300 \else

```

Tracking doesn't require a list, either.

```

2301 \MT@ifstreq\MT@feat{tr}\relax%

```

```

2302      \MT@warning{I cannot find a \nameuse{MT@abbr@MT@feat} list
2303      for font\MessageBreak`MT@font'%
2304      \ifx\MT@context\empty\else\space(context: `MT@context')\fi.
2305      Switching off\MessageBreak\nameuse{MT@abbr@MT@feat} for this font}%
2306  }%
2307  \fi
2308  \@secondoftwo
2309  }%
2310 }

\MT@get@inh@list The inheritance lists are global (no context).
\MT@context 2311 \def\MT@get@inh@list{%
2312   \let\MT@context\empty
2313   \MT@get@listname{MT@feat @inh}%
2314   \MT@ifdefined@c@TF\MT@listname{%
2315     \MT@edef\MT@MT@feat @inh\name{MT@listname}%
2316     (debug) \MT@info@n{1}{... Using \nameuse{MT@abbr@MT@feat} inheritance list
2317     (debug)           `MT@listname'}%
2318     \MT@let@c@n\@tempc{MT@MT@feat @inh\MT@listname}%

    If the list is \empty, it has already been parsed.
2319     \ifx\@tempc\empty \else
2320     (debug) \MT@info@n{1}{parsing inheritance list ...}%

    The group is only required in case an input encoding is given.
2321     \begingroup
2322     \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak`MT@listname'}%
2323     \MT@set@inputenc{inh}%
2324     \expandafter\MT@inh@do\@tempc,\relax,%
2325     \MT@gl@et@c{MT@MT@feat @inh\MT@listname}\empty
2326     \endgroup
2327     \fi
2328     }%
2329     \MT@let@c{MT@MT@feat @inh\name}\@undefined
2330     }%
2331 }

```

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@ 2332 \def\MT@get@slot{%
2333   \escapechar`\\
2334   \let\MT@char@m@ne
2335   \MT@noresttrue

```

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

```
2336 \MT@toks=\expandafter{\@tempa}%
```

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

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

```

2337 \expandafter\MT@is@letter\@tempa\relax\relax
2338 \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.

```
2339 \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.

```
2340 \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
2341 \MT@is@symbol
```

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

```
2342 {\expandafter\MT@is@composite\@tempa\relax\relax}%
2343 \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.

```
2344 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2345 \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
2346 \fi
2347 \fi

2348 \let\MT@char\MT@char@
2349 \MT@get@slot@
2350 \escapechar\m@ne
2351 }
2352 </package>
```

$\backslash MT@get@slot@$

```
2353 <*pdfTeX-def|luatex-def|xetex-def>
2354 \def\MT@get@slot@{%
```

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

```
2355 <xetex-def> \ifnum\XeTeXfonttype\MT@font=\z@
2356 \ifnum\MT@char > \m@ne
```

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

```
2357 <*luatex-def>
2358 \ifnum\MT@char=47\relax
2359 \ifMT@noreset \else
2360 \@tempcnta=\directlua{
2361   local glyph = microtype.name_to_slot([[ \expandafter\@gobble\@tempa ]])
2362   if glyph then tex.write(glyph)
2363   else tex.write(-1)
2364   end
2365 } \relax
2366 \ifnum\@tempcnta<\z@
2367 \MT@warn@unknown
2368 \let\MT@char\m@ne
2369 \else
2370 \edef\MT@char{\the\@tempcnta}%
2371 <debug>\MT@info@n1{3}{> \the\MT@toks' is a glyph name (\the\@tempcnta)}%
2372 \fi
2373 \fi
2374 \else
2375 </luatex-def>
```

If the user has specified something like ‘fi’, or wanted to define a number but forgot to use three digits, we’ll have something left of the string. In this case, we issue a warning and forget the complete string.

```

2376     \ifMT@noreset \else
2377         \MT@warn@rest
2378     <pdfTeX-def|luatex-def> \let\MT@char\m@ne
2379     <xetex-def> \let\MT@char\@empty
2380     \fi
2381 <luatex-def> \fi
2382     \else
2383         \MT@warn@unknown
2384     \fi
2385 <*xetex-def>
2386     \else

```

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

```

2387     \ifnum\MT@char=47\relax
2388         \ifMT@noreset \edef\MT@char{U47}%
2389         \else
2390             \@tempcnta=\XeTeXglyphindex"\expandafter\@gobble\@tempa"\relax
2391             \ifnum\@tempcnta=z@
2392                 \MT@warn@unknown
2393                 \let\MT@char\@empty
2394             \else
2395                 \edef\MT@char{\@tempa\space}%
2396                 \edef\MT@char@{-\the\@tempcnta}%
2397 <debug>\MT@debug@n1{3}{> `the\MT@toks' is a glyph name (\the\@tempcnta)}%
2398             \fi
2399         \fi
2400     \else
2401         \ifnum\MT@char > \m@ne
2402             \ifMT@noreset

```

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

```

2403         \@tempcnta=\XeTeXcharglyph\MT@char\relax
2404         \ifnum\@tempcnta=z@
2405             \MT@info@missing@char
2406             \let\MT@char\@empty
2407         \else
2408 <debug>\MT@debug@n1{3}{> (glyph number: \the\@tempcnta,
2409 <debug> glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%
2410         \edef\MT@char{U\MT@char}%
2411         \fi
2412     \else
2413         \MT@warn@rest
2414         \let\MT@char\@empty
2415     \fi
2416     \else
2417         \MT@warn@unknown
2418         \let\MT@char\@empty
2419     \fi
2420 \fi
2421 \fi
2422 </xetex-def>
2423 }
2424 </pdfTeX-def|luatex-def|xetex-def>

```

This is the lua function to translate glyph name into Unicode number, stolen from luaotfload (file otfl-font-dum.lua).

```

2425 <*luafile>
2426 function microtype.name_to_slot(name)

```

```

2427   if fonts then
2428     local tfmdata = fonts.ids[font.current()]
2429     if tfmdata and tfmdata.shared then
2430       local otfddata = tfmdata.shared.otfddata
2431       local unicode = otfddata.luatex.unicodes[name]
2432       return unicode and (type(unicode) == "number" and unicode or unicode[1])
2433     end
2434   end
2435 end
2436
2437 </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 2438 <*pdfTeX-def|luatex-def|xetex-def>
2439 \def\MT@max@char
2440 <pdfTeX-def> {127 }
2441 <luatex-def|xetex-def> {1114111 }
2442 \def\MT@max@slot
2443 <pdfTeX-def> {255 }
2444 <luatex-def|xetex-def> {1114111 }
2445 </pdfTeX-def|luatex-def|xetex-def>

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

2446 <*package>
2447 \newif\ifMT@noest

2448 \def\MT@is@letter#1#2\relax{%
2449   \ifcat a\noexpand#1\relax
2450     \edef\MT@char@{\number`#1}%
2451     \ifx\#2\%
2452       <debug>\MT@edinfo@n1{3}{> `the\MT@toks' is a letter (\MT@char@)}%
2453       \else
2454         \MT@noestfalse
2455       \fi
2456     \else
2457       \ifcat !\noexpand#1\relax
2458         \edef\MT@char@{\number`#1}%
2459       <debug>\MT@edinfo@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
2460       \ifx\#2\%
2461         \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
2462       \else
2463         \MT@noestfalse
2464         \expandafter\MT@is@number#1#2\relax\relax
2465       \fi
2466     \fi
2467   \fi
2468 }

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

2469 \def\MT@is@number#1#2#3\relax{%
2470   \ifx\relax#3\relax \else
2471     \ifx\relax#2\relax \else
2472       \MT@noesttrue
2473       \if#1"\relax
2474         \def\x{\uppercase{\edef\MT@char@{\number#1#2#3}}}\x
2475       <debug>\MT@edinfo@n1{3}{> ... a hexadecimal number: \MT@char@}%
2476       \else
2477         \if#1'\relax
2478         \def\MT@char@{\number#1#2#3}%
2479       <debug>\MT@edinfo@n1{3}{> ... an octal number: \MT@char@}%

```



```

2480         \else
2481         \MT@ifint{#1#2#3}{%
2482             \def\MT@char@{\number#1#2#3}%
2483 (debug)\MT@info@n1{3}{> ... a decimal number: \MT@char@}%
2484         }\MT@noestfalse
2485         \fi
2486         \fi
2487         \ifnum\MT@char@ > \MT@max@slot
2488             \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
2489             \let\MT@char@\m@ne
2490         \fi
2491         \fi
2492     \fi
2493 }

```

`\MT@is@active` Expand an active character. (This was completely broken in v1.7, and only worked by chance before.) We `\set@display@protect` to translate, e.g., Å into `\ "A`, that is to whatever it is defined in the `inputenc` encoding file.

Unfortunately, the (older) `inputenc` definitions prefer the protected/generic variants (e.g., `\copyright` instead of `\textcopyright`), which our parser won't be able to understand. (I'm fed up now, so you have to complain if you really, really want to be able to write '©' instead of `\textcopyright`, thus rendering your configuration files unportable.)

Unicode characters (`inputenc/utf8,utf8x`) are also supported.

```

2494 \def\MT@is@active#1#2\@nil{%
2495     \ifnum\catcode`#1 = \active
2496         \begingroup
2497             \set@display@protect
2498             \let\IeC\@firstofone
2499             \let\@inpenc@undefined@\MT@undefined@char

```

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

```

2500     \def\UTFviii@defined##1{\ifx ##1\relax
2501         \MT@undefined@char{utf8}\else\expandafter ##1\fi}%

```

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

```

2502     \MT@ifdefined@c@T\PrerenderUnicode
2503     {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%
2504     \edef\x{\endgroup
2505         \def\noexpand\@tempa{\@tempa}%

```

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

```

2506     \MT@toks={\the\MT@toks\space(= \@tempa)}%
2507     }%
2508     \x
2509     \fi
2510 }

```

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

```

2511 \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.

```

2512 \def\MT@is@symbol{%
2513     \expandafter\def\expandafter\MT@char\expandafter
2514         {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%
2515     \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2516         \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax

```

```
2517 \ifnum\MT@char@ < \z@
```

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

```
2518 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
2519 \fi
2520 }
```

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

```
\MT@charstring 2521 \begingroup
2522 \catcode`\=/\z@
2523 /MT@map@tlist@n{/CHARLEX}/@makeother
2524 /lowercase{%
2525 /def/x{/endgroup
2526 /def/MT@charstring{\CHAR"%
2527 /def/MT@is@char##1\CHAR"##2##3##4/relax{%
2528 /ifx/relax##4/relax
2529 /ifMT@xunicode
2530 /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
2531 /relax/relax/relax/relax/relax
2532 /fi
2533 /else
2534 /ifx/relax##1/relax
2535 /if##3\relax
2536 /edef/MT@char@{/number"##2}%
2537 /MT@ifstreq/MT@charstring{##3##4}/relax/MT@norestfalse
2538 /else
2539 /edef/MT@char@{/number"##2##3}%
2540 /MT@ifstreq/MT@charstring{##4}/relax/MT@norestfalse
2541 /fi
2542 <debug> /MT@dinfo@n1{3}{>~/the/MT@toks' is a \char (/MT@char@)}%
2543 /fi
2544 /fi
2545 }%
```

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

```
\MT@strip@prefix 2546 /def/MT@charxstring{\CHAR "%
\MT@is@charx 2547 /def/MT@strip@prefix##1>##2/relax{##2}%
2548 /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
2549 /ifx/relax##1/relax
2550 /ifx/relax##6/relax/else
2551 /edef/MT@char@{/number"##2##3##4##5}%
2552 /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@norestfalse
2553 <debug> /MT@dinfo@n1{3}{>~/the/MT@toks' is a xunicode \char (/MT@char@)}%
2554 /fi
2555 /fi
2556 }%
2557 }%
2558 }
2559 /x
```

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

```
2560 \def\MT@is@composite#1#2\relax{%
2561 \ifx\#2\\\else
```

Again, we construct a control sequence, this time of the form: `\\<encoding>\<accent>-<character>`, e.g., `\\T1"-a`, which we then expand once to see if it is a letter (if it has been defined by `\DeclareTextComposite`). This should be robust, finally, especially, since we also `\detokenize` the input instead of only `\stringifying` it. Thus, we will die gracefully even on wrong Unicode input without `utf8`.

```
2562 \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
2563 \string\csname\MT@encoding\endcsname
2564 \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%
```

```
2565 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
```

Again, xunicode.

```
2566 \ifnum\MT@char@ < \z@
2567 \ifMT@xunicode
2568 \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
2569 \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
2570 \MT@char\MT@charxstring\relax\relax\relax\relax\relax
2571 \fi
2572 \fi
2573 \fi
2574 }
```

[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 2575 \def\MT@set@list@name{%
2576 \edef\MT@curr@list@name{\@nameuse{MT@abbr@}\MT@feat} \list\noexpand\MessageBreak
2577 ~\@nameuse{MT@\MT@feat @c@name}}%
2578 }
```

`\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.

```
2579 \def\MT@warn@ascii{%
2580 \MT@warning@nl{Character ~\the\MT@toks' (= \MT@char@)
2581 is outside of ASCII range.\MessageBreak
2582 You must load the ~inputenc' package before using\MessageBreak
2583 8-bit characters in \MT@curr@list@name}%
2584 }
```

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

```
2585 \def\MT@warn@number@too@large#1{%
2586 \MT@warning@nl{%
2587 Number #1 in encoding ~\MT@encoding' too large!\MessageBreak
2588 Ignoring it in \MT@curr@list@name}%
2589 }
```

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

```
2590 \def\MT@warn@rest{%
2591 \MT@warning@nl{%
2592 Unknown slot number of character\MessageBreak~\the\MT@toks'%
2593 \MT@warn@maybe@inputenc\MessageBreak
2594 in font encoding ~\MT@encoding'.\MessageBreak
2595 Make sure it's a single character\MessageBreak
}
```

```

2596      (or a number) in \MT@curr@list@name}%
2597 }

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

2598 \def\MT@warn@unknown{%
2599   \MT@warning@nl{%
2600     Unknown slot number of character\MessageBreak`the\MT@toks'%
2601     \MT@warn@maybe@inputenc\MessageBreak
2602     in font encoding `\'MT@encoding' in \MT@curr@list@name}%
2603 }

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

2604 \def\MT@warn@maybe@inputenc{%
2605   \MT@ifdefined@n@T
2606   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}%
2607   { (input encoding `\'@nameuse
2608   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}')}%
2609 }
```

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`, `pifont` (loaded by `hyperref`), `tipa`, and probably many more. Furthermore, we had to include a hack for the `IEEEtran` class which loads all fonts in the class file itself (to fine tune inter-word spacing), and the `memoir` class, too. To cut this short: it seemed to get out of hand, and I decided that it would be better to use `\pickup@font` and decide for ourselves whether we've already seen that font. I hope the overhead isn't too large.

`\MT@font@list` We use a comma separated list.

```
\MT@font 2610 \let\MT@font@list\empty
2611 \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`.

```
2612 </package>
2613 <*package|letterspace>
2614 <plain>\MT@requires@latex2{
2615 \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`.

```
2616 \@ifpackageloaded{CJK}{%
2617 \ifpackagelater{CJK}{2006/10/17}% 4.7.0
2618 {\def\MT@orig@pickupfont{\CJK@ifundefined{CJK@plane}}}%
2619 {\def\MT@orig@pickupfont{\@ifundefined{CJK@plane}}}%
2620 \g@addto@macro\MT@orig@pickupfont
2621 {\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).

```
2622 \@ifpackageloaded{CJKutf8}%
2623 {\@ifpackagelater{CJKutf8}{2008/05/22}% 4.8.0
2624 {\ifpdf\expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}%
2625 {\@firstoftwo}}%
2626 {\@firstoftwo}%
2627 {\g@addto@macro\MT@orig@pickupfont{%
2628 \expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2629 \define@newfont\else\xdef\font@name{%
2630 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
2631 {\g@addto@macro\MT@orig@pickupfont{%
2632 \expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2633 \define@newfont\def\CJK@temp{v}%
2634 \ifx\CJK@temp\CJK@plane
2635 \expandafter\ifx\csname CJK@cmap@\f@family\CJK@plane\endcsname\relax
2636 \else\csname CJK@cmap@\f@family\CJK@plane\endcsname\fi
2637 \else \CJK@addcmmap\CJK@plane \fi
2638 \else\xdef\font@name{%
2639 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
2640 }{%
2641 \def\MT@orig@pickupfont{\expandafter\ifx\font@name\relax\define@newfont\fi}%
2642 }%
```

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

```
2643 \ifx\pickup@font\MT@orig@pickupfont \else
2644 \MT@warning@n1{%
2645 Command \string\pickup@font\space is not defined as expected.%
2646 \MessageBreak Patching it anyway. Some things may break%
2647 <*package>
2648 .\MessageBreak Double-check whether micro-typography is indeed%
2649 \MessageBreak applied to the document.%
2650 \MessageBreak (Hint: Turn on `verbose' mode)%
2651 </package>
2652 }%
2653 \fi
```

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

2654 `\g@addto@macro\pickup@font{\begingroup}%`

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

```
2655 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
2656 \g@addto@macro\pickup@font{%
2657   \escapechar\m@ne
2658 }*package>
2659 <debug> \global\MT@inannottrue
2660 <debug> \MT@glet\MT@pdf@annot\@empty
2661 <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.

```
2662 \MT@let@cn\MT@font{MT@subst@expandafter\string\font@name}%
2663 \ifx\MT@font\relax
2664   \let\MT@font\font@name
2665 \else
2666   \ifx\MT@font\font@name \else
2667 <debug> \MT@addto@annot{= substituted with \MT@font}%
2668   \MT@register@subst@font
2669   \fi
2670 \fi
2671 \MT@setupfont
2672 }*package>
2673 <letterspace> \MT@tracking
2674 \endgroup
2675 }%
2676 }*package>
```

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

2677 `\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.

```
2678 \g@addto@macro\do@subst@correction
2679 { \edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
2680   \MT@glet@nc{MT@subst@expandafter\string\font@name}\MT@font}%
```

`\add@accent` Inside `\add@accent`, we have to disable microtype's setup, since the grouping in `\MT@orig@add@accent` the patched `\pickup@font` would break the accent if different fonts are used for the base character and the accent. Fortunately, L^AT_EX takes care that the fonts used for the `\accent` are already set up, so that we cannot be overlooking them.

```
2681 \let\MT@orig@add@accent\add@accent
2682 \def\add@accent#1#2{%
2683   \let\pickup@font\MT@orig@pickupfont
2684   \MT@orig@add@accent{#1}{#2}%
2685   \let\pickup@font\MT@pickupfont
2686 }%
2687 }*package>
2688 }
2689 <plain>}\relax
2690 }*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.
2691 `\def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}`

`\MT@register@font` Register the current font.
2692 `\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).
2693 `\def\MT@register@subst@font{\MT@exp@one@n\MT@in@clist\font@name\MT@font@list`
2694 `\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.

2695 `\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.

2696 `\def\MT@check@font@cx{%`
2697 `\MT@if@true`
2698 `\MT@map@clist@c\MT@active@features{%`
2699 `\expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font`
2700 `\csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname`
2701 `\ifMT@inlist@`
2702 `\MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax`
2703 `\else`
2704 `\MT@if@false`
2705 `\fi`
2706 `%`
2707 `\ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi`
2708 `}`

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

2709 `\def\MT@register@subst@font@cx{%`
2710 `\MT@map@clist@c\MT@active@features{%`
2711 `\expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name`
2712 `\csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname`
2713 `\ifMT@inlist@ \else`
2714 `\MT@exp@cs\MT@xadd`
2715 `{MT@##1\csname MT@##1@context\endcsname font@list}%`
2716 `{\font@name,}%`
2717 `\fi`
2718 `%`
2719 `}`

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

2720 `\def\MT@register@font@cx{%`
2721 `\MT@map@clist@c\MT@active@features{%`
2722 `\MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else`
2723 `\MT@exp@cs\MT@xadd`
2724 `{MT@##1\csname MT@##1@context\endcsname font@list}%`
2725 `{\MT@font,}%`
2726 `\def\@tempa{##1}%`
2727 `\MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list`
2728 `\fi`
2729 `%`
2730 `}`

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

```
2731 \def\MT@maybe@rem@from@list#1{%
2732   \MT@ifstreq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
2733     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
2734     \MT@font \csname MT@\@tempa @#1font@list\endcsname
2735   }%
2736 }
```

`\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.

```
2737 \def\microtypecontext#1{\MT@addto@setup{\microtypecontext{#1}}}%
2738 \MT@addto@setup{%
2739   \DeclareRobustCommand\microtypecontext[1]{%
2740     \MT@setup@contexts
2741     \let\MT@reset@context\relax
```

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

```
2742   \let\glb@currsiz@empty
2743   \setkeys{MTC}{#1}%
2744   \selectfont
2745   \MT@reset@context
2746 }%
2747 }
```

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

```
2748 \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@`

```
2749 \def\MT@reset@context@{%
2750   \MT@vinfo{<<< Resetting contexts\on@line
2751   <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
2752   <debug> \MT@tr@context/\MT@kn@context/\MT@sp@context
2753   }%
2754   \selectfont
2755 }
```

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

```
2756 \def\MT@setup@contexts{%
2757   \MT@map@clist@c\MT@active@features
2758   {\MT@gl@et@c\MT@##1@font@list}\MT@font@list}%
2759   \MT@gl@et\MT@check@font\MT@check@font@cx
2760   \MT@gl@et\MT@register@font\MT@register@font@cx
2761   \MT@gl@et\MT@register@subst@font\MT@register@subst@font@cx
2762   \MT@gl@et\MT@setup@contexts\relax
2763 }
```

Define context keys.

```
2764 \MT@map@clist@c\MT@features@long{%
2765   \define@key{MTC}{#1}[]{}%
2766   \edef\@tempb{\@nameuse{MT@rbba#1}}%
2767   \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
2768   \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).

```
2769   \MT@ifempty{##1}{\def\MT@val{0}}{\def\MT@val{##1}}%
2770   \MT@exp@cs@ifx{MT@\@tempb @context}\MT@val
2771   <debug> \MT@dinfo{1}{>>> no change of #1 context: `'\MT@val'}%
2772   \else
```



```

2773 \MT@vinfo{>>> Changing #1 context to `~\MT@val'\MessageBreak\on@line
2774 <debug> \space(previous: ~\@nameuse{MT@\@tempb @context}')%
2775 }%
2776 \def\MT@reset@context{\aftergroup\MT@reset@context}%

```

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

```

2777 \MT@glet@nn{MT@reset@\@tempb @codes}{MT@reset@\@tempb @codes}%

```

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

```

2778 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
2779 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
2780 \ifMT@inlist@ \else
2781 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{~\MT@val}}%
2782 <debug> \MT@dinfo{1}{||| added #1 context: ~\@nameuse{MT@\@tempb @doc@contexts}}%
2783 \fi
2784 \MT@edef@n{MT@\@tempb @context}{~\MT@val}%
2785 \fi
2786 \fi
2787 }%
2788 }

```

\MT@pr@context Initialise the contexts.

```

\MT@ex@context 2789 \MT@exp@one@n\MT@map@clist@n{~\MT@features,n1}}%

```

```

\MT@tr@context 2790 \MT@def@n{MT@#1@context}{@}%

```

```

\MT@sp@context 2791 \MT@def@n{MT@#1@doc@contexts}{~{~}}%

```

```

2792 }

```

```

\MT@kn@context 2793 \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.

```

2794 \def\DeclareMicrotypeSet{%
2795 \@ifstar
2796 \MT@DeclareSetAndUseIt
2797 \MT@DeclareSet
2798 }

```

\MT@DeclareSet

```

2799 \newcommand\MT@DeclareSet[3]{}{%
2800 \KV@sp@def\@tempa{#1}%
2801 \MT@ifempty\@tempa{%
2802 \MT@map@clist@c\MT@features{~\MT@declare@sets{##1}{#2}{#3}}}%
2803 }{%
2804 \MT@map@clist@c\@tempa{%
2805 \KV@sp@def\@tempa{##1}%
2806 \MT@ifempty\@tempa\relax{%
2807 \MT@is@feature{set declaration ~#2'}{%
2808 \MT@exp@one@n\MT@declare@sets
2809 {\csname MT@rbba@\@tempa\endcsname}{#2}{#3}%
2810 }%
2811 }%
2812 }}%
2813 }%
2814 }

```

\MT@DeclareSetAndUseIt

```
2815 \newcommand\MT@DeclareSetAndUseIt[3] [] {%
2816   \MT@DeclareSet[#1]{#2}{#3}%
2817   \UseMicrotypeSet[#1]{#2}%
2818 }
```

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

```
2819 \let\MT@curr@set@name\@empty
```

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

```
2820 \def\MT@declare@sets#1#2#3{%
2821   \KV@sp@def\MT@curr@set@name{#2}%
2822   \MT@ifdefined@n@T{MT@#1@set@@\MT@curr@set@name}{%
2823     \MT@warning{Redefining \@nameuse{MT@abbr@#1} set ~\MT@curr@set@name'}%
2824     \MT@gl@et@nc{MT@#1@list@size@\MT@curr@set@name}\@empty
2825   }%
2826   \MT@gl@et@nc{MT@#1@set@@\MT@curr@set@name}\@empty
2827   <debug>\MT@din@fo{1}{declaring \@nameuse{MT@abbr@#1} set ~\MT@curr@set@name'}%
2828   \setkeys{MT@#1@set}{#3}%
2829 }
```

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

```
2830 \def\MT@define@set@key@#1#2{%
2831   \define@key{MT@#2@set}{#1} [] {%
2832     \MT@gl@et@nc{MT@#2@list@#1@\MT@curr@set@name}\@empty
2833     \MT@map@cl@ist@n{##1}{%
2834       \KV@sp@def\MT@val{###1}%
2835       \MT@get@high@level{#1}%
```

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

```
2836   \MT@exp@two@n@g@addto@macro
2837     {\csname MT@#2@list@#1@\MT@curr@set@name\expandafter\endcsname}%
2838     {\MT@val,}%
2839   }%
```

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

```
2840   \expandafter\g@addto@macro\expandafter\MT@font@sets
2841     {\csname MT@#2@list@#1@\MT@curr@set@name\endcsname
2842     <debug>\MT@din@fo@n1{1}{-- #1: \@nameuse{MT@#2@list@#1@\MT@curr@set@name}}}%
2843   }%
2844 }
```

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

```
2845 \def\MT@get@high@level#1{%
2846   \expandafter\MT@test@ast\MT@val*\@nil\relax{%
```

And ‘family = *’ will become \familydefault.

```
2847   \MT@ifempty\@tempa{\def\@tempa{#1}}\relax
2848   \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.

```
2849   }%
2850 }
```

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

```
2851 \def\MT@test@ast#1*#2\@nil{%
2852   \def\@tempa{#1}%
2853   \MT@ifempty{#2}%
2854 }
```

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

```
2855 \let\MT@font@sets\empty
2856 \def\MT@fix@font@set#1{%
2857   \xdef#1{#1}%
2858   \ifMT@fontspec
2859     \xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%
2860   \fi
2861   \global\@onelevel@sanitize#1%
2862 }
```

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

```
2863 \def\MT@define@set@key@size#1{%
2864   \define@key{MT@#1@set}{size}[]{}%
2865   \MT@map@cliston{##1}{%
2866     \KV@esp@def\MT@val{###1}%
2867     \expandafter\MT@get@range\MT@val--\@nil
2868     \ifx\MT@val\relax \else
2869       \MT@exp@cs\MT@xadd
2870       {MT@#1list@size@MT@curr@set@name}%
2871       {{\MT@lower}{\MT@upper}\relax}}%
2872   \fi
2873 }%
2874 <debug>\MT@dinfol{1}{-- size: \@nameuse{MT@#1list@size@MT@curr@set@name}}%
2875 }%
2876 }
```

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 2877 \def\MT@get@range#1-#2-#3\@nil{%
2878   \MT@ifempty{#1}{%
2879     \MT@ifempty{#2}{%
2880       \let\MT@val\relax
2881     }%
2882     \def\MT@lower{0}%
2883     \def\MT@val{#2}%
2884     \MT@get@size
2885     \edef\MT@upper{\MT@val}%
2886   }%
2887 }%
2888 \def\MT@val{#1}%
2889 \MT@get@size
2890 \ifx\MT@val\relax \else
2891   \edef\MT@lower{\MT@val}%
2892   \MT@ifempty{#2}{%
2893     \MT@ifempty{#3}%
2894     {\def\MT@upper{-1}}%
```

2048 pt is TeX's maximum font size.

```
2895   {\def\MT@upper{2048}}%
2896 }%
2897 \def\MT@val{#2}%
2898 \MT@get@size
2899 \ifx\MT@val\relax \else
2900   \MT@ifdim\MT@lower>\MT@val{%
2901     \MT@error{%
2902       Invalid size range (\MT@lower\space > \MT@val) in font set
```

```

2903         \MT@curr@set@name'.\MessageBreak Swapping sizes}}}%
2904         \edef\MT@upper{\MT@lower}%
2905         \edef\MT@lower{\MT@val}%
2906     }{%
2907         \edef\MT@upper{\MT@val}%
2908     }%
2909     \MT@ifdim\MT@lower=\MT@upper
2910     {\def\MT@upper{-1}}}%
2911     \relax
2912 \fi
2913 }%
2914 \fi
2915 }%
2916 }

```

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

```

2917 \def\MT@get@size{%
    A single star would mean \sizedefault, which doesn't exist, so we define it to be
    \normalsize.
2918     \if*\MT@val\relax
2919         \def\@tempa{\normalsize}%
2920     \else
2921         \MT@let@cn\@tempa{\MT@val}%
2922     \fi
2923     \ifx\@tempa\relax \else

```

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

```

2924     \begingroup
2925     \def\set@fontsize##1##2##3##4\@nil{\endgroup\def\MT@val{##2}}%
2926     \@tempa\@nil
2927 \fi

```

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

```

2928 \MT@ifdimen\MT@val{%
2929     \@defaultunits\@tempdima\MT@val pt\relax\@nnil
2930     \edef\MT@val{\strip@pt\@tempdima}%
2931 }{%
2932     \MT@warning{Could not parse font size `~\MT@val'\MessageBreak
2933         in font set `~\MT@curr@set@name'}%
2934     \let\MT@val\relax
2935 }%
2936 }

```

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

```

2937 \def\MT@define@set@key@font#1{%
2938     \define@key{MT@#1@set}{font}[]{}%
2939     \MT@get@nc{MT@#1list@font@\MT@curr@set@name}\@empty
2940     \MT@map@cliston{##1}{%
2941         \KV@sp@def\MT@val{###1}%
2942         \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
2943         \expandafter\MT@get@font\MT@val////\@nil
2944         \MT@exp@twoen@gaddto@macro
2945         {\csname MT@#1list@font@\MT@curr@set@name\expandafter\endcsname}%
2946         {\MT@val,%}%
2947     }%
2948     \expandafter\g@addto@macro\expandafter\MT@font@sets
2949     \csname MT@#1list@font@\MT@curr@set@name\endcsname
2950 (debug) \MT@edinfoenl{1}{-- font: \nameuse{MT@#1list@font@\MT@curr@set@name}}%

```

```
2951 }%
2952 }
```

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

```
2953 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
2954 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
2955 \ifx\MT@val\relax\def\MT@val{0}\fi
2956 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
2957 \let\MT@val\@tempb
2958 }
```

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

```
2959 \def\MT@get@font@#1#2#3#4#5#6{%
2960 \let\@tempb\@empty
2961 \def\MT@temp{#1/#2/#3/#4/#5}%
2962 \MT@get@axis{encoding}{#1}%
2963 \MT@get@axis{family}{#2}%
2964 \MT@get@axis{series}{#3}%
2965 \MT@get@axis{shape}{#4}%
2966 \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
2967 \MT@ifempty{#5}{%
2968 \MT@warn@axis@empty{size}{\string\normalsize}%
2969 \def\MT@val{*}%
2970 }{%
2971 \def\MT@val{#5}%
2972 }%
2973 \MT@get@size
2974 }
```

`\MT@get@axis`

```
2975 \def\MT@get@axis#1#2{%
2976 \def\MT@val{#2}%
2977 \MT@get@highlevel{#1}%
2978 \MT@ifempty\MT@val{%
2979 \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
2980 \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
2981 }\relax
2982 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
2983 }
```

`\MT@warn@axis@empty`

```
2984 \def\MT@warn@axis@empty#1#2{%
2985 \MT@warning{#1 axis is empty in font specification\MessageBreak
2986 ~\MT@temp'. Using ~#2' instead}%
2987 }
```

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

```
2988 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
2989 \MT@define@set@key@{encoding}{#1}%
2990 \MT@define@set@key@{family}{#1}%
2991 \MT@define@set@key@{series}{#1}%
2992 \MT@define@set@key@{shape}{#1}%
2993 \MT@define@set@key@size{#1}%
2994 \MT@define@set@key@font{#1}%
2995 }
```

`\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.

```
2996 \renewcommand*\UseMicrotypeSet[2][{}]{%
2997 \KV@sp@def\@tempa{#1}%
2998 \MT@ifempty\@tempa{%
2999 \MT@map@clist@c\MT@features{{\MT@use@set{##1}{#2}}}%
3000 }{%
```

```

3001 \MT@map@clist@c\@tempa{%
3002 \KV@sp@def\@tempa{##1}%
3003 \MT@ifempty\@tempa\relax{%
3004 \MT@is@feature{activation of set `#2'}{%
3005 \MT@exp@one@n\MT@use@set
3006 {\csname MT@rbba@\@tempa\endcsname}{#2}%
3007 }%
3008 }%
3009 }}%
3010 }%
3011 }

```

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

```

\MT@ex@setname 3012 \def\MT@use@set#1#2{%
\MT@tr@setname 3013 \KV@sp@def\@tempa{#2}%
3014 \MT@ifdefined@n@TF{MT@#1@set@@\@tempa}{%
\MT@sp@setname 3015 \MT@xdef@n{MT@#1@setname}{\@tempa}%
\MT@kn@setname 3016 }{%
\MT@use@set 3017 \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
3018 \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
3019 }%
3020 \MT@error{%
3021 The \@nameuse{MT@abbr@#1} set `@\tempa' is undeclared.\MessageBreak
3022 Using set `@\nameuse{MT@#1@setname}' instead}{}%
3023 }%
3024 }

```

\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.

```

3025 \renewcommand*{\DeclareMicrotypeSetDefault}[2][]{%
3026 \KV@sp@def\@tempa{#1}%
3027 \MT@ifempty\@tempa{%
3028 \MT@map@clist@c\MT@features{{\MT@set@default@set{##1}{#2}}}%
3029 }{%
3030 \MT@map@clist@c\@tempa{%
3031 \KV@sp@def\@tempa{##1}%
3032 \MT@ifempty\@tempa\relax{%
3033 \MT@is@feature{declaration of default set `#2'}{%
3034 \MT@exp@one@n\MT@set@default@set
3035 {\csname MT@rbba@\@tempa\endcsname}{#2}%
3036 }%
3037 }%
3038 }}%
3039 }%
3040 }

```

\MT@default@pr@set

```

\MT@default@ex@set 3041 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 3042 \KV@sp@def\@tempa{#2}%
3043 \MT@ifdefined@n@TF{MT@#1@set@@\@tempa}{%
\MT@default@sp@set 3044 (debug)\MT@info{1}{declaring default \@nameuse{MT@abbr@#1} set `@\tempa'}%
\MT@default@kn@set 3045 \MT@xdef@n{MT@default@#1@set}{\@tempa}%
\MT@set@default@set 3046 }{%
3047 \MT@error{%
3048 The \@nameuse{MT@abbr@#1} set `@\tempa' is not declared.\MessageBreak
3049 Cannot make it the default set. Using set\MessageBreak `all' instead}{}%
3050 \MT@xdef@n{MT@default@#1@set}{all}%
3051 }%
3052 }

```

14.3.2 Variants and aliases

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

```
3053 \let\MT@variants\@empty
3054 \def\DeclareMicrotypeVariants{%
3055   \ifstar
3056     \MT@DeclareVariants
3057   {\let\MT@variants\@empty\MT@DeclareVariants}%
3058 }
```

`\MT@DeclareVariants`

```
3059 \def\MT@DeclareVariants#1{%
3060   \MT@map@clist@n{#1}{%
3061     \KV@esp@def\@tempa{##1}%
3062     \@onelevel@sanitize\@tempa
3063     \xdef\MT@variants{\MT@variants{\@tempa}}%
3064   }%
3065 }
```

`\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.

```
3066 \renewcommand*\DeclareMicrotypeAlias[2]{%
3067   \edef\@tempa{\zap@space#1 \@empty}%
3068   \edef\@tempb{\zap@space#2 \@empty}%
3069   \@onelevel@sanitize\@tempb
3070   \MT@ifdefined@n@T{MT@\@tempa @alias}{%
3071     \MT@warning{Alias font family '\@tempb' will override
3072       alias '\@nameuse{MT@\@tempa @alias}'\MessageBreak
3073       for font family '\@tempa'}}%
3074   \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.

```
3075 \MT@ifdefined@c@T{MT@family}%
3076 <debug>\MT@edinfo{1}{Activating alias font '\@tempb' for '\MT@family'}%
3077 \MT@glet{MT@familyalias}\@tempb
3078 }%
3079 }
```

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

```
3080 \def\LoadMicrotypeFile#1{%
3081   \edef\@tempa{\zap@space#1 \@empty}%
3082   \@onelevel@sanitize\@tempa
3083   \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3084   \ifMT@inlist@
3085     \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
3086   \else
3087     \MT@xadd\MT@file@list{\@tempa,}%
3088     \MT@begin@catcodes
3089     \InputIfFileExists{mt-\@tempa.cfg}{%
3090       \edef\MT@curr@file{mt-\@tempa.cfg}%
3091       \MT@vinfo{... Loading configuration file \MT@curr@file}%
3092     }{%
3093       \MT@warning{... Configuration file mt-\@tempa.cfg\MessageBreak
3094         does not exist}%
3095     }%
3096     \MT@end@catcodes
3097   \fi
3098 }
3099 </package>
3100 </package|letterspace>
```

14.3.3 Disabling ligatures

`\DisableLigatures` This is really simple now: we can re-use the set definitions of `\DeclareMicrotypeSet`; there can only be one set, which we'll call 'no ligatures'.

`\MT@nl@setname` The optional argument may be used to disable selected ligatures only.

```

\MT@nl@ligatures 3101 <*pdfTeX-def|luatex-def>
3102 <pdfTeX-def>\MT@requires@pdfTeX5{
3103 \def\DisableLigatures{%
3104 \MT@begin@catcodes
3105 \MT@DisableLigatures
3106 }
3107 \newcommand*\MT@DisableLigatures[2][]{%
3108 \MT@ifempty{#1}\relax{\gdef\MT@nl@ligatures{#1}}%
3109 \xdef\MT@active@features{\MT@active@features,nl}%
3110 \global\MT@noLigaturestrue
3111 \MT@declare@sets{nl}{no ligatures}{#2}%
3112 \gdef\MT@nl@setname{no ligatures}%
3113 \MT@end@catcodes
3114 }
3115 <pdfTeX-def>}{
3116 </pdfTeX-def|luatex-def>

If pdfTeX is too old, we throw an error.
3117 <*pdfTeX-def|xetex-def>
3118 \renewcommand*\DisableLigatures[2][]{%
3119 \MT@error{Disabling ligatures of a font is only possible\MessageBreak
3120 with pdfTeX version 1.30 or newer.\MessageBreak
3121 Ignoring \string\DisableLigatures}{%
3122 <pdfTeX-def> Upgrade
3123 <xetex-def> Use
3124 pdfTeX.}%
3125 }
3126 <pdfTeX-def>}
3127 </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.

```

3128 <*package>
3129 \def\DeclareMicrotypeBabelHook#1#2{%
3130 \MT@map@clist@n{#1}{%
3131 \KV@sp@def\@tempa{##1}%
3132 \MT@gdef@n{MT@babel@\@tempa}{#2}%
3133 }%
3134 }
3135 </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).

```

3136 <*pdfTeX-def|xetex-def|luatex-def>
3137 \def\SetProtrusion{%
3138 \MT@begin@catcodes

```



```

3139 \MT@SetProtrusion
3140 }

\MT@SetProtrusion    We want the catcodes to be correct even if this is called in the preamble.
\MT@pr@c@name 3141 \newcommand*\MT@SetProtrusion[3] [] {%
\MT@extra@context 3142 \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.
3143 \MT@set@named@keys{MT@pr@c}{#1}%
3144 <debug>\MT@info{1}{creating protrusion list `~\MT@pr@c@name'}%
3145 \def\MT@permutelist{pr@c}%
3146 \setkeys{MT@cfg}{#2}%

                    We have parsed the second argument, and can now define macros for all permuta-
                    tions of the font attributes to point to \MT@pr@c@<name>, ...
3147 \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.
3148 \MT@gdef@n{MT@pr@c@~\MT@pr@c@name}{#3}%
3149 \MT@end@catcodes
3150 }
3151 </pdfTeX-def|xetex-def|luatex-def>

\SetExpansion    \SetExpansion only differs in that it allows some extra options (stretch, shrink,
                    step, auto).
3152 <*pdfTeX-def|luatex-def>
3153 \def\SetExpansion{%
3154 \MT@begin@catcodes
3155 \MT@SetExpansion
3156 }

\MT@SetExpansion
\MT@ex@c@name 3157 \newcommand*\MT@SetExpansion[3] [] {%
\MT@extra@context 3158 \let\MT@extra@context\@empty
3159 \MT@set@named@keys{MT@ex@c}{#1}%
\MT@permutelist 3160 \MT@ifdefined@n@T{MT@ex@c@~\MT@ex@c@name @factor}{%
3161 \ifnum\csname MT@ex@c@~\MT@ex@c@name @factor\endcsname > \@m
3162 \MT@warning@n1{Expansion factor \number\@nameuse{MT@ex@c@~\MT@ex@c@name @factor}
3163 too large in list\MessageBreak `~\MT@ex@c@name'. Setting it to the
3164 maximum of 1000}%
3165 \MT@glet@c{MT@ex@c@~\MT@ex@c@name @factor}\@m
3166 \fi
3167 }%
3168 <debug>\MT@info{1}{creating expansion list `~\MT@ex@c@name'}%
3169 \def\MT@permutelist{ex@c}%
3170 \setkeys{MT@cfg}{#2}%
3171 \MT@permute
3172 \MT@gdef@n{MT@ex@c@~\MT@ex@c@name}{#3}%
3173 \MT@end@catcodes
3174 }
3175 </pdfTeX-def|luatex-def>

\SetTracking
3176 <*pdfTeX-def|luatex-def>
3177 \def\SetTracking{%
3178 \MT@begin@catcodes
3179 \MT@SetTracking
3180 }

\MT@SetTracking    Third argument may be empty.
3181 \newcommand*\MT@SetTracking[3] [] {%

```

```

3182 \let\MT@extra@context\@empty
3183 \MT@set@named@keys{MT@tr@ec}{#1}%
3184 <debug>\MT@dinfn{1}{creating tracking list `~\MT@tr@ec@name'}%
3185 \def\MT@permutelist{tr@ec}%
3186 \setkeys{MT@c@fg}{#2}%
3187 \MT@permute
3188 \KV@sp@def\@tempa{#3}%
3189 \MT@ifempty\@tempa\relax{%
3190   \MT@ifint\@tempa
3191     {\MT@gdef\MT@tr@ec@MT@tr@ec@name}{\@tempa}}%
3192     {\MT@warning{Value `~\@tempa' is not a number in\MessageBreak
3193       tracking set `~\MT@curr@set@name'}}}%
3194 \MT@end@catcodes
3195 }
3196 </pdfTeX-def|LaTeX-def>

```

\SetExtraSpacing

```

3197 <*pdfTeX-def>
3198 \def\SetExtraSpacing{%
3199   \MT@begin@catcodes
3200   \MT@SetExtraSpacing
3201 }

```

\MT@SetExtraSpacing

```

\MT@sp@c@name 3202 \newcommand*\MT@SetExtraSpacing[3][]{%
\MT@extra@context 3203   \let\MT@extra@context\@empty
3204   \MT@set@named@keys{MT@sp@c}{#1}%
\MT@permutelist 3205 <debug>\MT@dinfn{1}{creating spacing list `~\MT@sp@c@name'}%
3206   \def\MT@permutelist{sp@c}%
3207   \setkeys{MT@c@fg}{#2}%
3208   \MT@permute
3209   \MT@gdef\MT@sp@c@MT@sp@c@name}{#3}%
3210   \MT@end@catcodes
3211 }

```

\SetExtraKerning

```

3212 \def\SetExtraKerning{%
3213   \MT@begin@catcodes
3214   \MT@SetExtraKerning
3215 }

```

\MT@SetExtraKerning

```

\MT@kn@c@name 3216 \newcommand*\MT@SetExtraKerning[3][]{%
\MT@extra@context 3217   \let\MT@extra@context\@empty
3218   \MT@set@named@keys{MT@kn@c}{#1}%
\MT@permutelist 3219 <debug>\MT@dinfn{1}{creating kerning list `~\MT@kn@c@name'}%
3220   \def\MT@permutelist{kn@c}%
3221   \setkeys{MT@c@fg}{#2}%
3222   \MT@permute
3223   \MT@gdef\MT@kn@c@MT@kn@c@name}{#3}%
3224   \MT@end@catcodes
3225 }
3226 </pdfTeX-def>

```

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

\MT@options

```

3227 <*package>
3228 \def\MT@set@named@keys#1#2{%
3229   \def\x##1name=##2,##3\@nil{%
3230     \setkeys{#1}{name=##2}%
3231     \gdef\MT@options{##1##3}%
3232     \MT@rem@from@clist{name=}\MT@options
3233   }%
3234   \x#2,name=,\@nil

```

```

3235 \expandtwoargs\setkeys{#1}\MT@options
3236 }

```

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

```

3237 \def\MT@define@code@key#1#2{%
3238 \define@key{MT@#2}{#1}[]{%
3239 \@tempcnta=\@ne
3240 \MT@map@clist@n{##1}{%
3241 \KV@@sp@def\MT@val{###1}%

```

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

```

3242 \MT@get@highlevel{#1}%
3243 \MT@edef@n{MT@temp#1\the\@tempcnta}{\MT@val}%
3244 \advance\@tempcnta \@ne
3245 }%
3246 }%
3247 }

```

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

```

3248 \def\MT@define@code@key@family#1{%
3249 \define@key{MT@#1}{family}[]{%
3250 \@tempcnta=\@ne
3251 \MT@map@clist@n{##1}{%
3252 \KV@@sp@def\MT@val{###1}%
3253 \MT@get@highlevel{family}%
3254 \ifMT@fontspec
3255 \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
3256 \fi
3257 \MT@edef@n{MT@tempfamily\the\@tempcnta}{\MT@val}%
3258 \advance\@tempcnta \@ne
3259 }%
3260 }%
3261 }

```

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

```

3262 \def\MT@define@code@key@size#1{%
3263 \define@key{MT@#1}{size}[]{%
3264 \MT@map@clist@n{##1}{%
3265 \KV@@sp@def\MT@val{###1}%
3266 \expandafter\MT@get@range\MT@val--\@nil
3267 \ifx\MT@val\relax \else
3268 \MT@exp@cs\MT@xadd{MT@tempsize}%
3269 {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
3270 \fi
3271 }%
3272 }%
3273 }

```

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

```

3274 \def\MT@define@code@key@font#1{%
3275 \define@key{MT@#1}{font}[]{%
3276 \MT@map@clist@n{##1}{%
3277 \KV@@sp@def\MT@val{###1}%
3278 \MT@ifstreql\MT@val*{\def\MT@val{*/*/*/*}}\relax
3279 \expandafter\MT@get@font@and@size\MT@val///// \@nil
3280 \ifMT@fontspec
3281 \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
3282 \fi
3283 \MT@xdef@n{MT@MT@permutelist @\@tempb\MT@extra@context}%
3284 {\csname MT@MT@permutelist @name\endcsname}%
3285 (debug) \MT@edinfol{1}{initialising: use list for font \MT@val
3286 (debug) \ifx\MT@extra@context\@empty\else\MessageBreak
3287 (debug) (context: \MT@extra@context)\fi}%
3288 \MT@exp@cs\MT@xaddb

```

```

3289      {MT@MT@permutelist @\@tempb\MT@extra@context @sizes}%
3290      {{{\MT@eval}{\m@ne}{\MT@curr@set@name}}}%
3291    }%
3292  }%
3293 }

```

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

```

3294 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
3295   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
3296 }

3297 \MT@define@code@key{encoding}{cfg}
3298 \MT@define@code@key{family} {cfg}
3299 \MT@define@code@key{series} {cfg}
3300 \MT@define@code@key{shape} {cfg}
3301 \MT@define@code@key{size} {cfg}
3302 \MT@define@code@key{font} {cfg}

```

`\MT@define@opt@key`

```

3303 \def\MT@define@opt@key#1#2{%
3304   \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax{%
3305     \MT@xdefn{MT@#1@c@MT@curr@set@name @#2}{##1}}}%
3306 }

```

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

```

3307 \newcount\MT@listname@count
3308 \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`).

```

3309   \define@key{MT@#1@c}{name}[]{%
3310     \MT@ifempty{##1}%
3311     \MT@ifdefinedn@TF{MT@#1@c@MT@curr@file/\the\inputlineno}{%
3312       \global\advance\MT@listname@count\@ne
3313       \MT@edefn{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
3314         (\number\MT@listname@count)}%
3315     }{%
3316       \MT@edefn{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
3317     }%
3318   }{%
3319     \MT@edefn{MT@#1@c@name}{##1}%
3320     \MT@ifdefinedn@T{MT@#1@c@\csname MT@#1@c@name\endcsname}{%
3321       \MT@warning{Redefining \@nameuse{MT@abbr@#1} list ~\@nameuse{MT@#1@c@name}}%
3322     }%
3323   }%
3324   \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
3325 }%
3326 \MT@define@opt@key{#1}{load}%
3327 \MT@define@opt@key{#1}{factor}%
3328 \MT@define@opt@key{#1}{preset}%
3329 \MT@define@opt@key{#1}{inputenc}%

```

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

```

3330 \define@key{MT@#1@c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
3331 }

```

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.

```

3332 </package>
3333 <*pdfTeX-def|luatex-def>
3334 <pdfTeX-def>\MT@requires@pdfTeX7{
3335   \define@key{MT@ex@c}{context}[]{%

```

```

3336 \MT@ifempty{#1}\relax{%
3337 \MT@gllet\MT@copy@font\MT@copy@font@
3338 \def\MT@extra@context{#1}%
3339 }%
3340 }
3341 \MT@addto@setup{%
3342 \define@key{MT@ex@c}{context}[]{%
3343 \ifx\MT@copy@font\MT@copy@font@
3344 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
3345 \else
3346 \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
3347 Ignoring `context' key\on@line}%
3348 {Either move the settings inside the preamble,\MessageBreak
3349 or load the package with the `copyfonts' option.}%
3350 \fi
3351 }%
3352 }

```

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.

```

3353 \define@key{MT@pr@c}{context}[]{%
3354 \MT@ifempty{#1}\relax{%
3355 \MT@gllet\MT@copy@font\MT@copy@font@
3356 \def\MT@extra@context{#1}%
3357 }%
3358 }
3359 \MT@addto@setup{%
3360 \define@key{MT@pr@c}{context}[]{%
3361 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
3362 \ifx\MT@copy@font\MT@copy@font@\else
3363 \MT@warning@nl{If protrusion contexts don't work as expected,
3364 \MessageBreak load the package with the `copyfonts' option}%
3365 \fi
3366 }%
3367 }
3368 </pdfTeX-def|luatex-def>
3369 <*pdfTeX-def>
3370 }{
3371 \define@key{MT@ex@c}{context}[]{%
3372 \MT@error{Expansion contexts only work with pdfTeX 1.40.4\MessageBreak
3373 or later. Ignoring `context' key\on@line}%
3374 {Upgrade pdfTeX.}%
3375 }
3376 </pdfTeX-def>
3377 <*pdfTeX-def|xetex-def>
3378 \define@key{MT@pr@c}{context}[]{%
3379 \MT@error{Protrusion contexts only work with pdfTeX
3380 <pdfTeX-def> 1.40.4\MessageBreak or later.
3381 <xetex-def> \MessageBreak or luatex.
3382 Ignoring `context' key\on@line}%
3383 <pdfTeX-def> {Upgrade pdfTeX.}%
3384 <xetex-def> {Use pdfTeX or luatex.}%
3385 }
3386 </pdfTeX-def|xetex-def>
3387 <pdfTeX-def>

```

\MT@warn@nodim

```

3388 <*package>
3389 \def\MT@warn@nodim#1{%

```

```

3390 \MT@warning{'\@tempa' is not a dimension.\MessageBreak
3391 Ignoring it and setting values relative to\MessageBreak #1}%
3392 }
3393 </package>

```

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

```

3394 <(*pdfTeX-def|xetex-def|luatex-def)
3395 \define@key{MT@pr@c}{unit}[character]{%
3396 \MT@glet@c{MT@pr@c@MT@curr@set@name @unit}\@empty
3397 \def\@tempa{#1}%
3398 \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.

```

3399 \MT@ifdimen\@tempa
3400 {\MT@glet@c{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
3401 {\MT@warn@nodim{character widths}}%
3402 }%
3403 }
3404 </pdfTeX-def|xetex-def|luatex-def>

```

Tracking may only be relative to a dimension.

```

3405 <(*pdfTeX-def|luatex-def)
3406 \define@key{MT@tr@c}{unit}[1em]{%
3407 \MT@glet@c{MT@tr@c@MT@curr@set@name @unit}\@empty
3408 \def\@tempa{#1}%
3409 \MT@ifdimen\@tempa
3410 {\MT@glet@c{MT@tr@c@MT@curr@set@name @unit}\@tempa}%
3411 {\MT@warn@nodim{1em}%
3412 \MT@gdefn{MT@tr@c@MT@curr@set@name @unit}{1em}}%
3413 }
3414 </pdfTeX-def|luatex-def>

```

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

```

3415 <(*pdfTeX-def)
3416 \MT@map@clist@n{sp,kn}{%
3417 \define@key{MT@#1@c}{unit}[space]{%
3418 \MT@glet@c{MT@#1@c@MT@curr@set@name @unit}\@empty
3419 \def\@tempa{##1}%
3420 \MT@ifstreq\@tempa{character}\relax{%
3421 \MT@glet@c{MT@#1@c@MT@curr@set@name @unit}\m@ne
3422 \MT@ifstreq\@tempa{space}\relax{%
3423 \MT@ifdimen\@tempa
3424 {\MT@glet@c{MT@#1@c@MT@curr@set@name @unit}\@tempa}%
3425 {\MT@warn@nodim{width of space}}%
3426 }%
3427 }%
3428 }%
3429 }
3430 </pdfTeX-def>

```

The first argument to \SetExpansion accepts some more options.

```

3431 <(*pdfTeX-def|luatex-def)
3432 \MT@map@clist@n{stretch,shrink,step}{%
3433 \define@key{MT@ex@c}{#1}[]{%
3434 \MT@ifempty{##1}\relax{%
3435 \MT@ifint{##1}%

```

A space terminates the number.

```

3436 \MT@gdefn{MT@ex@c@MT@curr@set@name @#1}{##1}%
3437 }{%
3438 \MT@warning{%
3439 Value `##1' for option `#1' is not a number.\MessageBreak
3440 Ignoring it}%
3441 }%

```

```

3442 }%
3443 }%
3444 }
3445 \define@key{MT@ex@ec}{auto}[true]{%
3446   \def\@tempa{#1}%
3447   \csname if\@tempa\endcsname

Don't use autoexpand for pdfTeX version older than 1.20.
3448 \pdfTeX-def \MT@requires@pdfTeX4{%
3449   \MT@gdefon{MT@ex@ec\MT@curr@set@name @auto}{autoexpand}%
3450 \pdfTeX-def
3451 }{%
3452   \MT@warning{pdfTeX too old for automatic font expansion}%
3453 }
3454 \pdfTeX-def
3455 \else
3456 \pdfTeX-def \MT@requires@pdfTeX4{%
3457   \MT@gletenc{MT@ex@ec\MT@curr@set@name @auto}\empty
3458 \pdfTeX-def } \relax
3459 \fi
3460 }
3461 \pdfTeX-def|luatex-def

```

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

```

3462 \pdfTeX-def|luatex-def
3463 \MT@define@opt@key{tr}{spacing}
3464 \MT@define@opt@key{tr}{outerspacing}
3465 \MT@define@opt@key{tr}{outerkerning}

```

Which ligatures should be disabled?

```

3466 \define@key{MT@tr@ec}{noligatures}[]%
3467   {\MT@xdefon{MT@tr@ec\MT@curr@set@name @noligatures}{#1}}
3468 \define@key{MT@tr@ec}{outer spacing}[]{\setkeys{MT@tr@ec}{outerspacing={#1}}}
3469 \define@key{MT@tr@ec}{outer kerning}[]{\setkeys{MT@tr@ec}{outerkerning={#1}}}
3470 \define@key{MT@tr@ec}{no ligatures}[]{\setkeys{MT@tr@ec}{noligatures={#1}}}
3471 \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.

```

3472 \package
3473 \renewcommand*\DeclareCharacterInheritance[1][]{%
3474   \let\MT@extra@context\empty
3475   \let\MT@extra@inputenc\undefined
3476   \let\MT@inh@feat\empty
3477   \setkeys{MT@inh@}{#1}%
3478   \MT@begin@catcodes
3479   \MT@set@inh@list
3480 }

```

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

```

3481 \def\MT@set@inh@list#1#2{%
3482   \MT@ifempty\MT@inh@feat{%

```

```

3483 \MT@map@clist@c\MT@features{{\MT@declare@char@inh{##1}{#1}{#2}}}%
3484 }{%
3485 \MT@map@clist@c\MT@inh@feat{{%
3486 \KV@esp@def\@tempa{##1}%
3487 \MT@ifempty\@tempa\relax{%
3488 \MT@exp@one@n\MT@declare@char@inh
3489 {\csname MT@rbba@\@tempa\endcsname}{#1}{#2}%
3490 }%
3491 }}%
3492 }%
3493 \MT@end@catcodes
3494 }

```

The keys for the optional argument.

```

3495 \MT@map@clist@c\MT@features@long{%
3496 \define@key{MT@inh@}{#1}[]{\edef\MT@inh@feat{\MT@inh@feat#1,}}
3497 \define@key{MT@inh@}{inputenc}{\def\MT@extra@inputenc{#1}}

```

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

```

3498 \def\MT@declare@char@inh#1#2#3{%
3499 \MT@edef@n{MT@#1@inh@name}%
3500 {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
3501 \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
3502 \MT@ifdefined@c@T\MT@extra@inputenc{%
3503 \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
3504 <debug>\MT@info{1}{creating inheritance list ` \@nameuse{MT@#1@inh@name}'}%
3505 \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
3506 \def\MT@permute@list{#1@inh}%
3507 \setkeys{MT@inh}{#2}%
3508 \MT@permute
3509 }

```

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

```

3510 \MT@define@code@key{encoding}{inh}
3511 \MT@define@code@key{family}{inh}
3512 \MT@define@code@key{series}{inh}
3513 \MT@define@code@key{shape}{inh}
3514 \MT@define@code@key{size}{inh}
3515 \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`).

```

3516 \def\MT@inh@do#1,{%
3517 \ifx\relax#1\@empty \else
3518 \MT@inh@split #1==\relax
3519 \expandafter\MT@inh@do
3520 \fi
3521 }

```

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

```

3522 </package>
3523 <*pdfTeX-def|xetex-def|luatex-def>
3524 \def\MT@inh@split#1=#2=#3\relax{%
3525 \def\@tempa{#1}%
3526 \ifx\@tempa\@empty \else
3527 \MT@get@slot
3528 <pdfTeX-def|luatex-def> \ifnum\MT@char > \m@ne

```



```

3529 <xetex-def> \ifx\MT@char\empty\else
3530 \let\MT@val\MT@char
3531 \MT@map@clist@n{#2}{%
3532 \def\@tempa{##1}%
3533 \ifx\@tempa\empty \else
3534 \MT@get@slot
3535 <pdfTeX-def|luatex-def> \ifnum\MT@char > \m@ne
3536 <xetex-def> \ifx\MT@char\empty\else
3537 \MT@exp@cs\MT@xadd\MT@inh\MT@listname @\MT@val @}{\MT@char}}%
3538 \fi
3539 \fi
3540 }%
3541 <debug>\MT@info@n1{2}{children of #1 (\MT@val):
3542 <debug> \@nameuse\MT@inh\MT@listname @\MT@val @}}%
3543 \fi
3544 \fi
3545 }
3546 </pdfTeX-def|xetex-def|luatex-def>
3547 <*package>

```

14.3.7 Permutation

`\MT@permute` Calling `\MT@permute` will define commands for all permutations of the specified font attributes of the form `\MT@<list type>@/<encoding>/<family>/<series>/<shape>/<|*>` to be the expansion of `\MT@<list type>@name`, i.e., the name of the currently defined list.

`\MT@permute@`

`\MT@permute@@` Size ranges are held in a separate macro called `\MT@<list type>@/@sizes`, which in turn contains the respective *<list name>*s attached to the ranges.

`\MT@permute@@@`

`\MT@permute@@@`

```

3548 \def\MT@permute{%
3549 \let\MT@cnt@encoding\@ne
3550 \MT@permute@

  Undefine commands for the next round.

3551 \MT@map@tlist@n{<encoding>}{<family>}{<series>}{<shape>}}\MT@permute@reset
3552 \MT@glet\MT@tempsize\undefined
3553 }
3554 \def\MT@permute@{%
3555 \let\MT@cnt@family\@ne
3556 \MT@permute@@
3557 \MT@increment\MT@cnt@encoding
3558 \MT@ifdefined@n@T{MT@tempencoding\MT@cnt@encoding}%
3559 \MT@permute@
3560 }
3561 \def\MT@permute@@{%
3562 \let\MT@cnt@series\@ne
3563 \MT@permute@@@
3564 \MT@increment\MT@cnt@family
3565 \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
3566 \MT@permute@@
3567 }
3568 \def\MT@permute@@@{%
3569 \let\MT@cnt@shape\@ne
3570 \MT@permute@@@
3571 \MT@increment\MT@cnt@series
3572 \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
3573 \MT@permute@@@
3574 }
3575 \def\MT@permute@@@{%
3576 \MT@permute@@@
3577 \MT@increment\MT@cnt@shape
3578 \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
3579 \MT@permute@@@
3580 }

```

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

```

3581 \def\MT@permute@@@@{%
3582   \MT@permute@define{encoding}%
3583   \ifMT@document
3584     \ifx\MT@tempencoding\@empty \else
3585       \MT@ifdefined@n@TF{T@\MT@tempencoding}\relax
3586       {\expandafter\expandafter\expandafter\@gobble}%
3587     \fi
3588   \fi
3589   \MT@permute@@@@
3590 }
```

\MT@permute@@@@

```

3591 \def\MT@permute@@@@{%
3592   \MT@permute@define{family}%
3593   \MT@permute@define{series}%
3594   \MT@permute@define{shape}%
3595   \edef\@tempa{\MT@tempencoding
3596             /\MT@tempfamily
3597             /\MT@tempseries
3598             /\MT@tempshape
3599             /\MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

3600 \MT@ifstreq\@tempa{////}\relax{%
3601   \ifx\MT@tempencoding\@empty
3602     \MT@warning{%
3603       You have to specify an encoding for\MessageBreak
3604       \@nameuse{MT@abbr@MT@permutelist} list
3605       ` \@nameuse{MT@MT@permutelist @name}'.\MessageBreak
3606       Ignoring it}%
3607   \else
3608     \MT@ifdefined@c@TF\MT@tempsize{%

```

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

```

3609   \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%
3610     \MT@map@tlist@c\MT@tempsize\MT@check@rlist
3611   }%
3612   \MT@exp@cs\MT@xaddb
3613   {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
3614   \MT@tempsize
3615   <debug>\MT@info@n{1}{initialising: use list for font \@tempa,\MessageBreak
3616   <debug>       sizes: \csname MT@MT@permutelist @\@tempa\MT@extra@context
3617   <debug>       @sizes\endcsname}%
3618   }{%

```

Only one list can apply to a given combination.

```

3619   \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context}{%
3620     \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
3621       \@nameuse{MT@MT@permutelist @name}' will override list\MessageBreak
3622       \@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
3623       for font \@tempa'}%
3624   }%
3625   <debug>\MT@info@n{1}{initialising: use list for font \@tempa
3626   <debug>   \ifx\MT@extra@context\@empty\else\MessageBreak
3627   <debug>   (context: \MT@extra@context)\fi}%
3628   }%
3629   \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}%
3630   {\csname MT@MT@permutelist @name\endcsname}%
3631   \fi
3632   }%
3633 }
```

```

\MT@permute@define    Define the commands.
3634 \def\MT@permute@define#1{%
3635   \@tempcnta=\csname MT@cnt@#1\endcsname\relax
3636   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3637   {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
3638   {\MT@let@nc{MT@temp#1}\empty}%
3639 }

\MT@permute@reset    Reset the commands.
3640 \def\MT@permute@reset#1{%
3641   \@tempcnta=\@ne
3642   \MT@loop
3643   \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
3644   \advance\@tempcnta\@ne
3645   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3646   \iftrue
3647   \iffalse
3648   \MT@repeat
3649 }

\MT@check@rlist    For every new range item in \MT@tempsize, check whether it overlaps with ranges
                   in the existing list.
3650 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

\MT@check@rlist@    Define the current new range and ...
3651 \def\MT@check@rlist@#1#2#3{%
3652   \def\@tempb{#1}%
3653   \def\@tempc{#2}%
3654   \MT@iffalse
3655   \MT@exp@cs\MT@map@tlist@c
3656   {MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}%
3657   \MT@check@range
3658 }

\MT@check@range    ... recurse through the list of existing ranges.
3659 \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.
3660 \def\MT@check@range@#1#2#3{%
3661   \MT@ifdim{#2}=\m@ne{%
3662     \MT@ifdim\@tempc=\m@ne{%

• Both items are simple sizes.

3663     \MT@ifdim\@tempb={#1}\MT@iftrue\relax
3664     }{%

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

3665     \MT@ifdim\@tempb>{#1}\relax{%
3666       \MT@ifdim\@tempc>{#1}{%
3667         \MT@iftrue
3668         \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
3669       }\relax
3670     }%
3671   }%
3672 }{%
3673   \MT@ifdim\@tempc=\m@ne{%

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

3674   \MT@ifdim\@tempb<{#2}{%
3675     \MT@ifdim\@tempb<{#1}\relax\MT@iftrue

```

```

3676     }\relax
3677     }{%

```

- Both items are ranges.

```

3678     \MT@ifdim\@tempb<{#2}{%
3679     \MT@ifdim\@tempc>{#1}{%
3680     \MT@iftrue
3681     \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
3682     }\relax
3683     }\relax
3684     }%
3685     }%
3686     \ifMT@if@
3687     \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
3688     \@nameuse{MT@MT@permutelist @name}' will override\MessageBreak
3689     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.

```

3690     \expandafter\MT@tlist@break
3691     \fi
3692 }

```

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 3693 \newif\ifMT@opt@expansion
\ifMT@opt@DVI 3694 \newif\ifMT@opt@auto
3695 \newif\ifMT@opt@DVI

```

`\MT@optwarn@admissible` Some warnings.

```

3696 \def\MT@optwarn@admissible#1#2{%
3697   \MT@warning@nl{`#1' is not an admissible value for option\MessageBreak
3698   `#2'. Assuming `false'}%
3699 }

```

`\MT@optwarn@nan`

```

3700 </package>
3701 <*package|letterspace>
3702 <plain>\MT@requires@latex1{
3703 \def\MT@optwarn@nan#1#2{%
3704   \MT@warning@nl{Value `#1' for option `#2' is not a\MessageBreak number.
3705   Using default value of \number\@nameuse{MT@#2@default}}%
3706 }
3707 <plain>}\relax
3708 </package|letterspace>
3709 <*package>

```

`\MT@opt@def@set`

```

3710 \def\MT@opt@def@set#1{%
3711   \MT@ifdefined@n@TF{MT@\@tempb @set@{\MT@val}}{%
3712     \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
3713   }{%
3714     \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
3715     \MT@warning@nl{The #1 set `MT@val' is undeclared.\MessageBreak
3716     Using set \@nameuse{MT@\@tempb @setname}' instead}%
3717   }%
3718 }

```

expansion and protrusion may be true, false, compatibility, nocompatibility and/or a *<set name>*.

```

3719 \MT@map@clist@n{protrusion,expansion}{%
3720   \define@key{MT}{#1}[true]{%
3721     \csname MT@opt@#1true\endcsname
3722     \MT@map@clist@n{##1}{%
3723       \KV@@sp@def\MT@val{###1}%
3724       \MT@ifempty\MT@val\relax{%
3725         \csname MT@#1true\endcsname
3726         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
3727         \MT@ifstreq\MT@val{true}\relax
3728         {%
3729           \MT@ifstreq\MT@val{false}{%
3730             \csname MT@#1false\endcsname
3731           }{%
3732             \MT@ifstreq\MT@val{compatibility}{%
3733               \MT@let@nc{MT@\@tempb @level}\@ne
3734             }{%
3735               \MT@ifstreq\MT@val{nocompatibility}{%
3736                 \MT@let@nc{MT@\@tempb @level}\tw@
3737               }{%

```

If everything failed, it should be a set name.

```

3738       \MT@opt@def@set{#1}%
3739     }%
3740   }%
3741 }%
3742 }%
3743 }%
3744 }%
3745 }%
3746 }

```

activate is a shortcut for protrusion and expansion.

```

3747 \define@key{MT}{activate}[true]{%
3748   \setkeys{MT}{protrusion={#1}}%
3749   \setkeys{MT}{expansion={#1}}%
3750 }

```

spacing, kerning and tracking do not have a compatibility level.

```

3751 \MT@map@clist@n{spacing,kerning,tracking}{%
3752   \define@key{MT}{#1}[true]{%
3753     \MT@map@clist@n{##1}{%
3754       \KV@@sp@def\MT@val{###1}%
3755       \MT@ifempty\MT@val\relax{%
3756         \csname MT@#1true\endcsname
3757         \MT@ifstreq\MT@val{true}\relax
3758         {%
3759           \MT@ifstreq\MT@val{false}{%
3760             \csname MT@#1false\endcsname
3761           }{%
3762             \edef\@tempb{\csname MT@rbba@#1\endcsname}%
3763             \MT@opt@def@set{#1}%
3764           }%
3765         }%
3766       }%
3767     }%
3768   }%
3769 }

```

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

```

3770 \def\MT@def@bool@opt#1#2{%
3771   \define@key{MT}{#1}[true]{%

```

```

3772 \def\@tempa{##1}%
3773 \MT@ifstreq\@tempa{true}\relax{%
3774 \MT@ifstreq\@tempa{false}\relax{%
3775 \MT@optwarn@admissible{##1}{#1}%
3776 \def\@tempa{false}%
3777 }%
3778 }%
3779 #2%
3780 }%
3781 }

```

Boolean options that only set the switch.

```

3782 \MT@map@clist@n{draft,selected,babel}{%
3783 \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
3784 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotru}

```

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

```

3785 </package>
3786 (*pdfTeX-def|luatex-def|xetex-def)
3787 \MT@def@bool@opt{DVIPut}{%
3788 \csname if\@tempa\endcsname
3789 (*pdfTeX-def|luatex-def)
3790 \ifnum\pdfoutput>\z@ \MT@opt@DVITrue \fi
3791 \pdfoutput\z@
3792 \else
3793 \ifnum\pdfoutput<\@ne \MT@opt@DVITrue \fi
3794 \pdfoutput\@ne
3795 </pdfTeX-def|luatex-def>
3796 (xetex-def) \MT@warning@n1{Ignoring `DVIPut' option}%
3797 \fi
3798 }
3799 </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.

```

3800 (*package)
3801 \MT@def@bool@opt{defersetup}{%
3802 \csname if\@tempa\endcsname \else
3803 \AtEndOfPackage{%
3804 \MT@setup@
3805 \let\MT@setup@\empty
3806 \let\MT@addto@setup\@firstofone
3807 }%
3808 \fi
3809 }
3810 </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.

```

3811 (*pdfTeX-def|luatex-def)
3812 (pdfTeX-def)\MT@requires@pdfTeX7{
3813 \MT@def@bool@opt{copyfonts}{%
3814 \csname if\@tempa\endcsname
3815 \MT@gl@et\MT@copy@font\MT@copy@font@
3816 \else

```

```

3817 \MT@gllet\MT@copy@font\relax
3818 \fi
3819 }
3820 <pdfTeX-def>{}{
3821 </pdfTeX-def|luaTeX-def>
3822 <*pdfTeX-def|xetex-def>
3823 \MT@def@bool@opt{copyfonts}{%
3824 \csname if\@tempa\endcsname
3825 \MT@error
3826 <pdfTeX-def> {The pdfTeX version you are using is too old\MessageBreak
3827 <pdfTeX-def> to use the 'copyfonts' option}{Upgrade pdfTeX.}%
3828 <xetex-def> {The 'copyfonts' option does not work with xetex}
3829 <xetex-def> {Use pdfTeX or luaTeX instead.}%
3830 \fi
3831 }
3832 <pdfTeX-def>{}{
3833 </pdfTeX-def|xetex-def>

```

final is the opposite to draft.

```

3834 <*package>
3835 \MT@def@bool@opt{final}{%
3836 \csname if\@tempa\endcsname
3837 \MT@draftfalse
3838 \else
3839 \MT@drafttrue
3840 \fi
3841 }

```

For verbose output, we redefine \MT@vinfo.

```

3842 \define@key{MT}{verbose}[true]{%
3843 \let\MT@vinfo\MT@info@n1
3844 \def\@tempa{#1}%
3845 \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

3846 \MT@ifstreq\@tempa{errors}{%
3847 \let\MT@warning \MT@warn@err
3848 \let\MT@warning@n1\MT@warn@err
3849 }{%
3850 \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

3851 \MT@ifstreq\@tempa{silent}{%
3852 \let\MT@warning \MT@info
3853 \let\MT@warning@n1\MT@info@n1
3854 }{%
3855 \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
3856 }%
3857 }%
3858 }%
3859 }

```

Options with numerical keys: factor, stretch, shrink, step, letterspace.

```

3860 </package>
3861 <*package|letterspace>
3862 <plain>\MT@requires@latex1{
3863 \MT@map@clist@n{%
3864 <package> stretch,shrink,step,%
3865 letterspace}{%
3866 \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
3867 \def\@tempa{##1 }%

```

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

```

3868 \MT@ifint\@tempa
3869 {\MT@edef@n{MT@#1}{\@tempa}}%

```

```

3870      {\MT@optwarn@nan{##1}{#1}}%
3871    }%
3872  }
3873  \plain}\relax
3874  \package|letterspace
3875  \package

```

factor will define the protrusion factor only.

```

3876  \define@key{MT}{factor}{\MT@factor@default}{%
3877    \def\@tempa{#1}%
3878    \MT@ifint\@tempa
3879    {\edef\MT@pr@factor{\@tempa}}
3880    {\MT@optwarn@nan{#1}{factor}}%
3881  }

```

Unit for protrusion codes.

```

3882  \define@key{MT}{unit}{character}{%
3883    \def\@tempa{#1}%
3884    \MT@ifstreq\@tempa{character}\relax{%
3885      \MT@ifdimen\@tempa
3886      {\let\MT@pr@unit\@tempa}%
3887      {\MT@warning@nl{\@tempa' is not a dimension.\MessageBreak
3888        Ignoring it and setting values relative to\MessageBreak
3889        character widths}}%
3890    }%
3891  }

```

14.4.2 Loading the definition file

`\MT@endinput` Abort if no capable engine found.

```

3892  \let\MT@endinput\relax
3893  \ifx\MT@engine\relax
3894    \MT@warning@nl{You don't seem to be using either pdfTeX, luatex, or xetex.\MessageBreak
3895      '\MT@MT' only works with these engines.\MessageBreak
3896      I will quit now.}
3897    \MT@clear@options
3898  \else
3899    \input{microtype-MT@engine tex.def}
3900  \fi
3901  \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 .)

```

3902  \MT@protrusiontrue
3903  \package
3904  \pdfTeX-def|luatex-def
3905  \ifnum\pdfoutput<\@ne \else

```

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

```

3906  \pdfTeX-def \MT@requires@pdfTeX4{
3907    \MT@expansiontrue
3908    \MT@autottrue
3909  \pdfTeX-def }\relax
3910  \fi
3911  \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.

```

3912 (*package)
3913 \define@key{MT}{config}[]{\relax}
3914 \def\MT@get@config#1config=#2,#3\@nil{%
3915   \MT@ifempty{#2}%
3916   {\def\MT@config@file{\MT@MT.cfg}}%
3917   {\def\MT@config@file{#2.cfg}}%
3918 }
3919 \expandafter\expandafter\expandafter\MT@get@config
3920 \csname opt@\currname.\@currentx\endcsname,config=\@nil

```

Load the file.

```

3921 \IfFileExists{\MT@config@file}{%
3922   \MT@info{\Loading configuration file \MT@config@file}%
3923   \MT@begin@catcodes
3924   \let\MT@begin@catcodes\relax
3925   \let\MT@end@catcodes\relax
3926   \let\MT@curr@file\MT@config@file
3927   \input{\MT@config@file}%
3928   \endgroup
3929 }{\MT@warning{\MT@config@file}%
3930   Could not find configuration file '\MT@config@file'!\MessageBreak
3931   This will almost certainly cause undesired results.\MessageBreak
3932   Please fix your installation}%
3933 }

```

`\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).

```

3934 \def\MT@check@active@set#1{%
3935   \MT@ifdefined@n@TF{MT@#1@setname}{%
3936     \MT@info{\Using \nameuse{MT@abbr@#1} set '\nameuse{MT@#1@setname}'}%
3937   }{%
3938     \MT@ifdefined@n@TF{MT@default@#1@set}{%
3939       \MT@glenn{MT@#1@setname}{MT@default@#1@set}%
3940       \MT@info{\Using default \nameuse{MT@abbr@#1} set '\nameuse{MT@#1@setname}'}%
3941     }{%

```

If no default font set has been declared in the main configuration file, we use the (empty, non-existent) set '@', and issue a warning.

```

3942       \MT@gdefn{MT@#1@setname}{@}%
3943       \MT@warning{\No \nameuse{MT@abbr@#1} set chosen, no default set declared.
3944         \MessageBreak Using empty set}%
3945     }%
3946   }%
3947 }

```

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.

```
3948 \MT@ifdefined@c@T\MicroType@Hook{\MT@warning{%
3949   Command \string\MicroType@Hook\space is deprecated.\MessageBreak
3950   Use \string\Microtype@Hook\space instead}\MicroType@Hook}
3951 \MT@ifdefined@c@T\Microtype@Hook\Microtype@Hook
```

14.4.5 Changing options later

`\microtypesetup` Inside the preamble, `\microtypesetup` accepts the same options as the package (unless `defersetup=false`). In the document body, it accepts the options: protrusion, expansion, activate, tracking, spacing and kerning. Specifying font sets is not allowed.

```
3952 \def\microtypesetup{\setkeys{MT}}
3953 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
3954 /package
3955 (*pdfTeX-def|luatex-def|xetex-def)
3956 \def\MT@define@optionX#1#2{%
3957   \define@key{MTX}{#1}[true]{%
3958     \edef\@tempb{\csname MT@rbba@#1\endcsname}%
3959     \MT@map@clist@n{##1}{%
3960       \KV@sp@def\MT@val{###1}%
3961       \MT@ifempty\MT@val\relax{%
3962         \@tempcnta=\m@ne
3963         \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.

```
3964   \MT@checksetup{#1}{%
3965     \@tempcnta=\csname MT@@tempb @level\endcsname
3966     \MT@vinfo{Enabling #1
3967       (level \number\csname MT@@tempb @level\endcsname)\on@line}%
3968     }%
3969   }{%
3970     \MT@ifstreq\MT@val{false}{%
3971       \@tempcnta=\z@
3972       \MT@vinfo{Disabling #1\on@line}%
3973     }{%
3974       \MT@ifstreq\MT@val{compatibility}{%
3975         \MT@checksetup{#1}{%
3976           \@tempcnta=\@ne
3977           \MT@let@nc{MT@@tempb @level}\@ne
3978           \MT@vinfo{Setting #1 to level 1\on@line}%
3979         }%
3980       }{%
3981         \MT@ifstreq\MT@val{nocompatibility}{%
3982           \MT@checksetup{#1}{%

```

```

3983             \@tempcnta=\tw@
3984             \MT@let@nc{MT@\@tempb @level}\tw@
3985             \MT@vinfo{Setting #1 to level 2\on@line}%
3986         }%
3987         {\MT@error{Value `\'MT@val\' for key `\'#1\' not recognised}
3988          {Use any of `\'true\'', `\'false\'', `\'compatibility\' or
3989           `\'nocompatibility\''.}%
3990     }%
3991 }%
3992 }%
3993 }%
3994 \ifnum\@tempcnta>\m@ne
3995     #2\@tempcnta\relax
3996 \fi
3997 }%
3998 }%
3999 }%
4000 }

```

`\MT@checksetup` Test whether the feature wasn't disabled in the package options.

```

4001 \def\MT@checksetup#1{%
4002   \csname ifMT@#1\endcsname
4003   \expandafter\@firstofone
4004   \else
4005     \MT@error{You cannot enable #1 if it was disabled\MessageBreak
4006              in the package options}{Load microtype with #1 enabled.}%
4007     \expandafter\@gobble
4008   \fi
4009 }

4010 \MT@define@optionX{protrusion}\MT@protrudechars
4011 (/pdfTeX-def|luaTeX-def|xetex-def)
4012 (*pdfTeX-def|luaTeX-def)
4013 \MT@define@optionX{expansion}\MT@adjustspacing

```

`\MT@protrudechars`

```

\MT@adjustspacing 4014 \let\MT@protrudechars\pdfprotrudechars
4015 \let\MT@adjustspacing\pdfadjustspacing
4016 (/pdfTeX-def|luaTeX-def)
4017 (*xetex-def)
4018 \let\MT@protrudechars\XeTeXprotrudechars
4019 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
4020 (/xetex-def)

```

`\MT@define@optionX@` The same for tracking, spacing and kerning, which do not have a compatibility level.

```

4021 (*pdfTeX-def|luaTeX-def)
4022 (pdfTeX-def)\MT@requires@pdfTeX6{
4023 (luaTeX-def)\MT@requires@luaTeX3{
4024   \def\MT@define@optionX@#1#2{%
4025     \define@key{MTX}{#1}[true]{%
4026       \MT@map@clist@n{##1}{%
4027         \KV@sp@def\MT@val{###1}%
4028         \MT@ifempty\MT@val\relax{%
4029           \@tempcnta=\m@ne
4030           \MT@ifstreql\MT@val{true}{%
4031             \MT@checksetup{#1}%
4032             \@tempcnta=\@ne
4033             \MT@vinfo{Enabling #1\on@line}%
4034           }%
4035         }{%
4036           \MT@ifstreql\MT@val{false}{%
4037             \@tempcnta=\z@
4038             \MT@vinfo{Disabling #1\on@line}%
4039           }{\MT@error{Value `\'MT@val\' for key `\'#1\' not recognised}

```

```

4040             {Use either `true' or `false'}%
4041         }%
4042     }%
4043     \ifnum\@tempcnta>\m@ne
4044     #2\relax
4045     \fi
4046 }%
4047 }%
4048 }%
4049 }

```

We cannot simply let `\MT@tracking` relax, since this may select the already letter-spaced font instance.

```

4050 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
4051 \else \let\MT@tracking\MT@tracking@ \fi}
4052 \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
4053 \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
4054 \pdfappendkern \@tempcnta}
4055 }{
4056 </pdfTeX-def|luatex-def>
4057 <*pdfTeX-def|luatex-def|xetex-def>

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

4058 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
4059 <luatex-def>
4060 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
4061 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
4062 <pdfTeX-def>
4063 \define@key{MTX}{activate}[true]{%
4064 \setkeys{MTX}{protrusion={#1}}}%
4065 <pdfTeX-def|luatex-def> \setkeys{MTX}{expansion={#1}}}%
4066 }
4067 </pdfTeX-def|luatex-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.*)

```

4068 <*package>
4069 \let\MT@saved@setupfont\MT@setupfont
4070 \define@key{MTX}{disable}[]{%
4071 \MT@info{Inactivate `~\MT@MT' package}%
4072 \let\MT@setupfont\relax
4073 }
4074 \define@key{MTX}{enable}[]{%
4075 \MT@info{Reactivate `~\MT@MT' package}%
4076 \let\MT@setupfont\MT@saved@setupfont
4077 }
4078 </package>

```

14.4.6 Processing the options

`\MT@ProcessOptionsWithKV` Parse options.

```

4079 <*package|letterspace>
4080 <plain>\MT@requires@latex1{
4081 \def\MT@ProcessOptionsWithKV#1{%
4082 \let\@tempc\relax
4083 \let\MT@temp\@empty
4084 <plain> \MT@requires@latex2{
4085 \MT@map@c@list@c\@classoptionslist{%
4086 \def\CurrentOption{##1}%
4087 \MT@ifdefined@n@T{KV@#1@expandafter\MT@getkey\CurrentOption=\@nil}{%
4088 \edef\MT@temp{\MT@temp,\CurrentOption,}%
4089 \@expandtwoargs\@removeelement\CurrentOption
4090 \@unusedoptionlist\@unusedoptionlist

```

```

4091     }%
4092   }%
4093   \edef\MT@temp{\noexpand\setkeys{#1}%
4094     {\MT@temp\@ptionlist{\@currname.\@currentext}}}%

```

`explain` can handle package options.

```

4095 \begin{plain}
4096   {\edef\MT@temp{\noexpand\setkeys{#1}%
4097     {\csname usepkg@options@\usepkg@pkg\endcsname}}}
4098 \end{plain}
4099 \MT@temp
4100 \MT@clear@options
4101 }

```

`\MT@getkey` For key=val in class options.

```

4102 \def\MT@getkey#1=#2\@nil{#1}
4103 \MT@ProcessOptionsWithKV{MT}
4104 \begin{plain}\relax
4105 \end{package|letterspace}
4106 \end{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).

```

4107 \MT@addto@setup{%
4108 \ifMT@draft

```

We disable most of what we've just defined in the 4108 lines above if we are running in draft mode.

```

4109   \MT@warning@nl{'draft' option active.\MessageBreak
4110     Disabling all micro-typographic extensions.\MessageBreak
4111     This might lead to different line and page breaks}%
4112   \let\MT@setupfont\relax
4113   \renewcommand*\LoadMicrotypeFile[1]{}%
4114   \renewcommand*\microtypesetup[1]{}%
4115   \renewcommand*\microtypecontext[1]{}%
4116   \renewcommand*\lsstyle{}%
4117 \else
4118   \MT@setup@PDF
4119   \MT@setup@copies

```

Fix the font sets.

```

4120   \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
4121   \MT@setup@protrusion
4122   \MT@setup@expansion
4123   \MT@setup@tracking
4124   \MT@setup@wartracking
4125   \MT@setup@spacing
4126   \MT@setup@kerning
4127   \MT@setup@noligatures
4128 }
4129 \end{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!

```

4130 \begin{pdftex-def|luatex-def}
4131 \def\MT@setup@PDF{%

```

```

4132 \MT@info@nl{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
4133           \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
4134 }

```

\MT@setup@copies Working on font copies?

```

4135 \def\MT@setup@copies{%
4136   \ifx\MT@copy@font\relax\else \MT@info@nl{Using font copies for contexts}\fi
4137 }
4138 </pdfTeX-def|luatex-def>
4139 <*xetex-def>
4140 \let\MT@setup@PDF\relax
4141 \let\MT@setup@copies\relax
4142 </xetex-def>

```

\MT@setup@protrusion Protrusion.

```

4143 <*pdfTeX-def|xetex-def|luatex-def>
4144 \def\MT@setup@protrusion{%
4145   \ifMT@protrusion
4146     \edef\MT@active@features{\MT@active@features,pr}%
4147     \MT@protrudechars\MT@pr@level
4148     \MT@info@nl{Character protrusion enabled (level \number\MT@pr@level)%
4149               \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
4150               factor: \number\MT@pr@factor\fi
4151               \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
4152     \MT@check@active@set{pr}%
4153   \else
4154     \let\MT@protrusion\relax
4155     \MT@info@nl{No character protrusion}%
4156   \fi
4157 }
4158 </pdfTeX-def|xetex-def|luatex-def>

```

\MT@setup@expansion For DVI output, the user must have explicitly passed the expansion option to the package.

```

4159 <*pdfTeX-def|luatex-def>
4160 \def\MT@setup@expansion{%
4161   \ifnum\pdfoutput<\@ne
4162     \ifMT@opt@expansion \else
4163       \MT@expansionfalse
4164     \fi
4165   \fi
4166   \ifMT@expansion

```

Set up the values for font expansion: if stretch has not been specified, we take the default value of 20.

```

4167   \ifnum\MT@stretch=\m@ne
4168     \let\MT@stretch\MT@stretch@default
4169   \fi

```

If shrink has not been specified, it will inherit the value from stretch.

```

4170   \ifnum\MT@shrink=\m@ne
4171     \let\MT@shrink\MT@stretch
4172   \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.

```

4173   \ifnum\MT@step=\m@ne
4174 <pdfTeX-def> \MT@requires@pdfTeX6{%
4175   \def\MT@step{1 }%
4176 <*pdfTeX-def>

```

```

4177 }{%
4178   \ifnum\MT@stretch>\MT@shrink
4179     \ifnum\MT@shrink=\z@
4180       \@tempcnta=\MT@stretch
4181     \else
4182       \@tempcnta=\MT@shrink
4183     \fi
4184   \else
4185     \ifnum\MT@stretch=\z@
4186       \@tempcnta=\MT@shrink
4187     \else
4188       \@tempcnta=\MT@stretch
4189     \fi
4190   \fi
4191   \divide\@tempcnta 5\relax
4192   \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
4193   \edef\MT@step{\number\@tempcnta\space}%
4194 }%
4195 </pdfTeX-def>
4196 \fi
4197 \ifnum\MT@step=\z@
4198   \MT@warning@nl{The expansion step cannot be set to zero.\MessageBreak
4199     Setting it to one}%
4200   \def\MT@step{1 }%
4201 \fi

```

`\MT@auto` Automatic expansion of the font? This new feature of pdf \TeX 1.20 makes the *l^hz* programme really usable. It must be either ‘autoexpand’ or empty (or ‘1000’ for older versions of pdf \TeX).

```

4202 \let\MT@auto\empty
4203 \ifMT@auto
4204 <pdfTeX-def> \MT@requires@pdfTeX4{%

```

We turn off automatic expansion if output mode is DVI.

```

4205   \ifnum\pdfoutput<\@ne
4206     \ifMT@opt@auto
4207       \MT@error{%
4208         Automatic font expansion only works for PDF output.\MessageBreak
4209         However, you are creating a DVI file}
4210       {If you have created expanded fonts instances, remove `auto' from%
4211         \MessageBreak the package options. Otherwise, you have to switch
4212         off expansion.\MessageBreak completely.}%
4213     \fi
4214     \MT@autofalse
4215   \else
4216     \def\MT@auto{autoexpand}%
4217   \fi

```

Also, if pdf \TeX is too old.

```

4218 <*pdfTeX-def>
4219 }{%
4220   \MT@error{%
4221     The pdfTeX version you are using is too old for.\MessageBreak
4222     automatic font expansion}%
4223   {If you have created expanded fonts instances, remove `auto' from.\MessageBreak
4224     the package options. Otherwise, you have to switch off expansion.\MessageBreak
4225     completely, or upgrade pdfTeX to version 1.20 or newer.}%
4226   \MT@autofalse
4227   \def\MT@auto{1000 }%
4228 }%
4229 </pdfTeX-def>
4230 \else

```

No automatic expansion.

```

4231 <*pdfTeX-def>

```

```

4232 \MT@requires@pdftex4\relax{%
4233 \def\MT@auto{1000 }%
4234 }%
4235 </pdftex-def>
4236 \fi

```

Choose the appropriate macro for selected expansion.

```

4237 \ifMT@selected
4238 \let\MT@set@ex@codes\MT@set@ex@codes@s
4239 \else
4240 \let\MT@set@ex@codes\MT@set@ex@codes@n
4241 \fi

```

Filter out stretch=0, shrink=0, since it would result in a pdfTeX error.

```

4242 \ifnum\MT@stretch=\z@
4243 \ifnum\MT@shrink=\z@
4244 \MT@warning@n1{%
4245 Both the stretch and shrink limit are set to zero.\MessageBreak
4246 Disabling font expansion}%
4247 \MT@expansionfalse
4248 \fi
4249 \fi
4250 \fi
4251 \ifMT@expansion
4252 \edef\MT@active@features{\MT@active@features,ex}%
4253 \MT@adjustspacing\MT@ex@level
4254 \MT@info@n1{\ifMT@auto A\else Non-a\fi utomatic font expansion enabled
4255 (level \number\MT@ex@level),\MessageBreak
4256 stretch: \number\MT@stretch, shrink: \number\MT@shrink,
4257 step: \number\MT@step, \ifMT@selected\else non-\fi selected}%

```

\MT@check@step Check whether stretch and shrink are multiples of step.

```

4258 \def\MT@check@step##1{%
4259 \@tempcnta=\csname MT@##1\endcsname
4260 \divide\@tempcnta \MT@step
4261 \multiply\@tempcnta \MT@step
4262 \ifnum\@tempcnta=\csname MT@##1\endcsname\else
4263 \MT@warning@n1{The ##1 amount is not a multiple of step.\MessageBreak
4264 The effective maximum ##1 is \the\@tempcnta\space
4265 (step \number\MT@step)}%
4266 \fi
4267 }%
4268 \MT@check@step{stretch}%
4269 \MT@check@step{shrink}%
4270 \MT@check@active@set{ex}%

```

Inside \showhyphens, font expansion should be disabled.

```

4271 \CheckCommand*\showhyphens[1]{\setbox0\vbox{%
4272 \color@begingroup\everypar{}\parfillskip\z@skip
4273 \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
4274 \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%

```

\showhyphens I wonder why it's defined globally (in ltfsbas.dtx)?

```

4275 \gdef\showhyphens##1{\setbox0\vbox{%
4276 \color@begingroup\pdfadjustspacing\z@\everypar{}\parfillskip\z@skip
4277 \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
4278 \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
4279 \else
4280 \let\MT@expansion\relax
4281 \MT@info@n1{No font expansion}%
4282 \fi
4283 }
4284 </pdftex-def| luatex-def>
4285 <*xetex-def>
4286 \def\MT@setup@expansion{%

```



```

4287 \ifMT@expansion
4288 \ifMT@opt@expansion
4289 \MT@error{Font expansion does not work with xetex}
4290 {Use pdftex or luatex instead.}%
4291 \fi
4292 \fi
4293 }
4294 </xetex-def>

```

`\MT@setup@tracking` Tracking, spacing and kerning.

```

4295 <*pdftex-def>|<luatex-def>
4296 <pdftex-def>\MT@requires@pdftex6{%
4297 <luatex-def>\MT@requires@luatex3{%
4298 \def\MT@setup@tracking{%
4299 \ifMT@tracking
4300 \edef\MT@active@features{\MT@active@features,tr}%
4301 \MT@info@nl{Tracking enabled}%
4302 \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

4303 \ifMT@protrusion\else\MT@protrudechars\@ne\fi
4304 \else
4305 \let\MT@tracking\relax
4306 \MT@info@nl{No adjustment of tracking}%
4307 \fi
4308 }
4309 </pdftex-def>|<luatex-def>
4310 <*pdftex-def>

```

`\MT@setup@spacing`

```

4311 \def\MT@setup@spacing{%
4312 \ifMT@spacing
4313 \edef\MT@active@features{\MT@active@features,sp}%
4314 \pdfadjustinterwordglue\@ne
4315 \MT@info@nl{Adjustment of interword spacing enabled}%

```

The ragged2e package sets interword spaces to a fixed value without glue. microtype's modifications can therefore have undesired effects. Therefore, we issue a warning.

```

4316 \MT@with@package@T{ragged2e}{%
4317 \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
4318 Adjustment of interword spacing may lead to\MessageBreak
4319 undesired results when used with `ragged2e'.\MessageBreak
4320 In this case, disable the `spacing' option}%
4321 }%
4322 \MT@check@active@set{sp}%
4323 \else
4324 \let\MT@spacing\relax
4325 \MT@info@nl{No adjustment of interword spacing}%
4326 \fi
4327 }

```

`\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

```

4328 \def\MT@setup@spacing@check{%
4329 \ifMT@spacing
4330 \ifMT@babel \else
4331 \ifnum\sfcode`. > 1500
4332 \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
4333 \MT@warning@nl{%
4334 \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: ddtbaj\$rob\$1@online.de

```

4335             interword spacing will disable it. You might want\MessageBreak
4336             to add `~\backslashchar\MT@MT context{spacing=nonfrench}'\MessageBreak
4337             to your preamble}%
4338         }%
4339     \fi
4340 \fi
4341 \fi
4342 }

\MT@setup@kerning
4343 \def\MT@setup@kerning{%
4344     \ifMT@kerning
4345         \edef\MT@active@features{\MT@active@features,kn}%
4346         \pdfprependkern\@ne
4347         \pdfappendkern\@ne
4348         \MT@info@nl{Adjustment of character kerning enabled}%
4349         \MT@check@active@set{kn}%
4350     \else
4351         \let\MT@kerning\relax
4352         \MT@info@nl{No adjustment of character kerning}%
4353     \fi
4354 }
4355 </pdfTeX-def>

\MT@error@doesnt@work    If pdfTeX is too old, we disable tracking, spacing and kerning, and throw an error
                        message. We also switch the features off for LuaTeX and XeTeX.
4356 <pdfTeX-def|luatex-def>{}{
4357 <*luatex-def>
4358     \def\MT@setup@tracking{%
4359         \ifMT@tracking
4360             \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
4361             or newer. Switching it off}{Upgrade luatex.}%
4362             \MT@trackingfalse
4363             \MT@let@nc{MT@tracking}\relax
4364         \else
4365             \MT@info@nl{No adjustment of tracking (luatex too old)}%
4366         \fi
4367     }
4368 }
4369 </luatex-def>
4370 <*pdfTeX-def|xetex-def|luatex-def>
4371     \def\MT@error@doesnt@work#1{%
4372         \csname ifMT@#1\endcsname
4373         \MT@error{The #1 feature only works with pdfTeX 1.40\MessageBreak
4374         or newer. Switching it off}
4375 <pdfTeX-def>         {Upgrade pdfTeX.}%
4376 <luatex-def|xetex-def>         {Use pdfTeX instead.}%
4377         \csname MT@#1false\endcsname
4378         \MT@let@nc{MT@#1}\relax
4379     \else
4380         \MT@info@nl{No adjustment of #1%
4381 <pdfTeX-def>         \space(pdfTeX too old)%
4382     }%
4383     \fi
4384 }
4385 <pdfTeX-def|xetex-def> \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
4386 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
4387 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
4388 <pdfTeX-def>{}
4389 </pdfTeX-def|xetex-def|luatex-def>

\MT@setup@warntracking
4390 <letterspace>\MT@addto@setup
4391 <pdfTeX-def|luatex-def>\def\MT@setup@warntracking

```

`\MT@warn@tracking@DVI` We issue a warning, when letterspacing in DVI mode, since it will probably not work. We also switch on protrusion if it isn't already, to compensate for the letterspacing kerns.

```

4392 <pdfTeX-def|LaTeX-def|letterspace>
4393 {%
4394   \ifnum\pdfoutput<\@ne
4395     \def\MT@warn@tracking@DVI{%
4396       \MT@warning@n1{%
4397         You are using tracking/letterspacing in DVI mode.\MessageBreak
4398         This will probably not work, unless the post-\MessageBreak
4399         processing program (dvips, dvipdfm(x), ...) is\MessageBreak
4400         able to create the virtual fonts on the fly}%
4401       \MT@gllet\MT@warn@tracking@DVI\relax
4402     }%
4403   \else
4404     \def\MT@warn@tracking@DVI{%
4405       \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
4406       \MT@gllet\MT@warn@tracking@DVI\relax
4407     }%
4408   \fi
4409   \ifnum\MT@letterspace=\m@ne
4410     \let\MT@letterspace\MT@letterspace@default
4411   \else
4412     \MT@ls@too@large\MT@letterspace
4413   \fi
4414 }
4415 </pdfTeX-def|LaTeX-def|letterspace>
4416 <XeTeX-def>\let\MT@setup@warntracking\relax

```

`\MT@setup@noligatures` `\DisableLigatures` is only admissible in the preamble, therefore we can now disable the corresponding macro, if it was never called.

```

4417 <pdfTeX-def|LaTeX-def>
4418 \def\MT@setup@noligatures{%
4419 <pdfTeX-def> \MT@requires@pdfTeX5{%
4420   \ifMT@noligatures \else
4421     \let\MT@noligatures\relax
4422   \fi
4423 <pdfTeX-def> }\relax
4424 }
4425 </pdfTeX-def|LaTeX-def>
4426 <XeTeX-def>\let\MT@setup@noligatures\relax

```

Remove the leading comma in `\MT@active@features`, and set the document switch to true.

```

4427 <package>
4428 \MT@addto@setup{%
4429   \ifx\MT@active@features\@empty \else
4430     \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
4431   \fi
4432   \MT@documenttrue
4433 }

```

`\MT@set@babel@context` Interaction with babel.

```

4434 \def\MT@set@babel@context#1{%
4435   \MT@ifdefined@n@TF{MT@babel@#1}{%
4436     \MT@vinfo{*** Changing to language context `#1'\MessageBreak\on@line}%
4437     \expandafter\MT@exp@one@n\expandafter\microtypecontext
4438     \csname MT@babel@#1\endcsname
4439   }{%
4440     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
4441   }%
4442 }

```

`\MT@shorthandoff` Active characters can only be switched off if babel isn't loaded after microtype.

```

4443 \ifpackageloaded{babel}{
4444   \def\MT@shorthandoff#1#2{%
4445     \MT@info@n1{Switching off #1 babel's active characters (#2)}%
4446     \shorthandoff{#2}}
4447 }{
4448   \def\MT@shorthandoff#1#2{%
4449     \MT@error{You must load `babel' before `MT@MT'}
4450     {Otherwise, `MT@MT' cannot switch off #1 babel's\MessageBreak
4451       active characters.}}
4452 }
```

We patch the language switching commands to enable language-dependent setup.

```

4453 \MT@addto@setup{%
4454   \ifMT@babel
4455     \ifpackageloaded{babel}{%
4456       \MT@info@n1{Redefining babel's language switching commands}%
4457       \let\MT@orig@select@language\select@language
4458       \def\select@language#1{%
4459         \MT@orig@select@language{#1}%
4460         \MT@set@babel@context{#1}%
4461       }%
4462       \let\MT@orig@foreign@language\foreign@language
4463       \def\foreign@language#1{%
4464         \MT@orig@foreign@language{#1}%
4465         \MT@set@babel@context{#1}%
4466       }%
4467       \ifMT@kerning
```

Disable French babel's active characters.

```

4468     \MT@if@false
4469     \MT@with@babel@and@T{french} \MT@if@true
4470     \MT@with@babel@and@T{frenchb} \MT@if@true
4471     \MT@with@babel@and@T{francais} \MT@if@true
4472     \MT@with@babel@and@T{canadien} \MT@if@true
4473     \MT@with@babel@and@T{acadian} \MT@if@true
4474     \ifMT@if@MT@shorthandoff{French}{:;!}\fi
```

Disable Turkish babel's active characters.

```

4475     \MT@if@false
4476     \MT@with@babel@and@T{turkish} \MT@if@true
4477     \ifMT@if@MT@shorthandoff{Turkish}{:!=}\fi
4478     \fi
```

In case babel was loaded before microtype:

```

4479     \MT@set@babel@context\languagename
4480   }{%
4481     \MT@warning@n1{You did not load the babel package.\MessageBreak
4482       The `babel' option won't have any effect}%
4483   }%
4484   \fi
4485 }
```

Now we close the `\fi` from `\ifMT@draft`.

```

4486 \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.

```

4487   \selectfont}
```

`\MT@curr@file` This is the current file (hopefully with the correct extension).

```

4488 \edef\MT@curr@file{\jobname.tex}
```

Finally, execute the setup macro at the end of the preamble, and empty it (the combine class calls it repeatedly).

```
4489 </package>
4490 <*package|letterspace>
4491 <plain>\MT@requires@latex1{
4492 \AtBeginDocument{\MT@setup@ \MT@gl@et\MT@setup@ \@empty}
4493 <plain>}\relax
4494 </package|letterspace>
```

Must come at the very, very end.

```
4495 <package>\MT@ifdefined@c@T\MT@setup@spacing@check
4496 <package> { \AtBeginDocument{\MT@setup@spacing@check}}
```

Restore catcodes.

```
4497 <package|letterspace>\MT@restore@catcodes
```

And here we end the lua file.

```
4498 <luafile>end
```

That was that.

15 Configuration files

Let's now write the font configuration files.

```
4499 <*config>
4500
```

15.1 Font sets

We first declare some sets in the main configuration file.

```
4501 <*m-t>
4502 %%% -----
4503 %%% FONT SETS
4504
4505 \DeclareMicrotypeSet{all}
4506 { }
4507
4508 \DeclareMicrotypeSet{allmath}
4509 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TS1,OML,OMS,U} }
4510
4511 \DeclareMicrotypeSet{alltext}
4512 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2} }
4513
4514 \DeclareMicrotypeSet{basicmath}
4515 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,OML,OMS},
4516   family   = {rm*,sf*},
4517   series    = {md*},
4518   size      = {normalsize,footnotesize,small,large}
4519 }
4520
4521 \DeclareMicrotypeSet{basictext}
4522 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2},
4523   family   = {rm*,sf*},
4524   series    = {md*},
4525   size      = {normalsize,footnotesize,small,large}
4526 }
4527
```

```

4528 \DeclareMicrotypeSet{smallcaps}
4529   { encoding = {OT1,T1,T2A,Ly1,OT4,QX,T5,TS1,EU1,EU2},
4530     shape     = {sc*}
4531   }
4532
4533 \DeclareMicrotypeSet{footnotesize}
4534   { encoding = {OT1,T1,T2A,Ly1,OT4,QX,T5,TS1,EU1,EU2},
4535     size      = {-small}
4536   }
4537
4538 \DeclareMicrotypeSet{scriptsize}
4539   { encoding = {OT1,T1,T2A,Ly1,OT4,QX,T5,TS1,EU1,EU2},
4540     size      = {-footnotesize}
4541   }
4542
4543 \DeclareMicrotypeSet{normal font}
4544   { font = */*/*/*/* }
4545

```

The default sets.

```

4546 %%% -----
4547 %%% DEFAULT SETS
4548
4549 \DeclareMicrotypeSetDefault[protrusion]{alltext}
4550 \DeclareMicrotypeSetDefault[expansion]{basictext}
4551 \DeclareMicrotypeSetDefault[spacing]{basictext}
4552 \DeclareMicrotypeSetDefault[kerning]{alltext}
4553 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
4554

```

15.2 Font variants and aliases

```

4555 %%% -----
4556 %%% FONT VARIANTS AND ALIASES
4557

```

These are the variants I happen to be using (expert encoding, oldstyle numerals, swashes, alternative, display, inferior and superior numerals):

```

4558 \DeclareMicrotypeVariants{x,j,w,a,d,0,1}
4559

```

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.

```

4560 \ifMT@fontspec
4561 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
4562 \else
4563 \DeclareMicrotypeAlias{lmr}{cmr} % lmodern
4564 \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.

```

4565 \DeclareMicrotypeAlias{lmsy}{cmsy}
4566 \DeclareMicrotypeAlias{lmm}{cmm}
4567 \DeclareMicrotypeAlias{aer}{cmr} % ae

```

```

4568 \DeclareMicrotypeAlias{zer}{cmr} % zefonts
4569 \DeclareMicrotypeAlias{cmor}{cmr} % eco
4570 \DeclareMicrotypeAlias{hfor}{cmr} % hfoldsty

```

The packages pxfonts and txfonts fonts inherit Palatino and Times settings respectively, also the TeX Gyre fonts Pagella and Termes (formerly: qfonts).

```

4571 \DeclareMicrotypeAlias{pxr}{ppl} % pxfonts
4572 \DeclareMicrotypeAlias{qpl}{ppl} % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

The ‘FPL Neu’ fonts, a ‘re-implementation’ of Palatino.

```

4573 \DeclareMicrotypeAlias{fp9x}{pplx} % FPL Neu
4574 \DeclareMicrotypeAlias{fp9j}{pplj} % "
4575 \DeclareMicrotypeAlias{txr}{ptm} % txfonts
4576 \DeclareMicrotypeAlias{qtm}{ptm} % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The OpenType versions:

```

4577 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino Linotype}
4578 \DeclareMicrotypeAlias{Palatino LT Std}{Palatino Linotype}
4579 \DeclareMicrotypeAlias{Palatino}{Palatino Linotype}
4580 \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.

```

4581 \DeclareMicrotypeAlias{zeur}{eur} % Euler VM
4582 \DeclareMicrotypeAlias{zeus}{eus} % "

```

MicroPress’s Charter version (chmath).

```

4583 \DeclareMicrotypeAlias{chr}{bch} % CH Math

```

The mathdesign package provides math fonts matching Bitstream Charter and URW Garamond.

```

4584 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
4585 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond

```

The garamondx package, an extension of URW Garamond, providing small caps and oldstyle figures.

```

4586 \DeclareMicrotypeAlias{zgmX}{ugm} % garamondx
4587 \DeclareMicrotypeAlias{zgmj}{ugm} % "
4588 \DeclareMicrotypeAlias{zgmi}{ugm} % "
4589 \DeclareMicrotypeAlias{zgmq}{ugm} % "

```

URW Letter Gothic is similar enough to Bitstream Letter Gothic to share the configuration.

```

4590 \DeclareMicrotypeAlias{ulg}{blg} % URW LetterGothic -> Bitstream LetterGothic12Pitch

```

Euro symbol fonts, to save some files.

```

4591 \DeclareMicrotypeAlias{zpeus}{zpeu} % Adobe Euro sans -> serif
4592 \DeclareMicrotypeAlias{eurosans}{zpeu} % Adobe Euro sans -> serif
4593 \DeclareMicrotypeAlias{euroitcs}{euroitc} % ITC Euro sans -> serif
4594

```

15.3 Interaction with babel

Contexts that are to be set when switching to a language.

```

4595 %%% -----
4596 %%% INTERACTION WITH THE `babel' PACKAGE
4597
4598 \DeclareMicrotypeBabelHook
4599 {english,UKenglish,british,USenglish,american}
4600 {kerning=, spacing=nonfrench}
4601
4602 \DeclareMicrotypeBabelHook
4603 {french,français,acadian,canadien}
4604 {kerning=french, spacing=}

```

```

4605
4606 \DeclareMicrotypeBabelHook
4607   {turkish}
4608   {kerning=turkish, spacing=}
4609

```

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`ed symbols have to be specified numerically. Of course, numerical identification is possible in any other case, too.

8-bit characters are also admissible, provided they have been declared in the input encoding file. They should, however, only be used in private configuration files, where the proper input encoding is guaranteed, or else in combination with the `'inputenc'` key.

15.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not œ for O .

```

4610 </m-t>
4611 <*m-t|zpeu|mys>
4612 %%% -----
4613 %%% CHARACTER INHERITANCE
4614
4615 </m-t|zpeu|mys>
4616 <*m-t>

```

15.5.1 OT1

Glyphs that should possibly inherit settings on one side only: 012 (`'fi'` ligature), 013 (`'fl'`), 014 (`'ffi'`), 015 (`'ffl'`), Æ , æ , œ , ø .

```

4617 \DeclareCharacterInheritance
4618   { encoding = OT1 }
4619   { f = {011}, % ff
4620     i = {\i},
4621     j = {\j},
4622     0 = {\0},
4623     o = {\o}

```



```

4624   }
4625

```

15.5.2 T1

Candidates here: 028 (‘fi’), 029 (‘fl’), 030 (‘ffi’), 031 (‘ffl’), 156 (‘IJ’ ligature, since L^AT_EX 2005/12/01 accessible as \IJ), 188 (‘ij’, \ij), Æ, æ, Œ, œ.

```

4626 \DeclareCharacterInheritance
4627   { encoding = T1 }
4628   { A = {\`A,\^A,\^A,\~A,\"A,\r A,\k A,\u A},
4629     a = {\`a,\^a,\^a,\~a,\"a,\r a,\k a,\u a},
4630     C = {\`C,\c C,\v C},
4631     c = {\`c,\c c,\v c},
4632     D = {\v D,\DH},
4633     d = {\v d,\dj},
4634     E = {\`E,\^E,\^E,\"E,\k E,\v E},
4635     e = {\`e,\^e,\^e,\"e,\k e,\v e},
4636     f = {027}, % ff
4637     G = {\u G},
4638     g = {\u g},
4639     I = {\`I,\^I,\^I,\"I,\.I},
4640     i = {\`i,\^i,\^i,\"i,\i},
4641     j = {\j},
4642     L = {\L,\^L,\v L},
4643     l = {\l,\^l,\v l},
4644     N = {\`N,\~N,\v N},
4645     n = {\`n,\~n,\v n},
4646     O = {\0,\`0,\^0,\^0,\~0,\"0,\H 0},
4647     o = {\o,\`o,\^o,\^o,\~o,\"o,\H o},
4648     R = {\`R,\v R},
4649     r = {\`r,\v r},
4650     S = {\`S,\c S,\v S,\SS},
4651     s = {\`s,\c s,\v s},
4652     T = {\c T,\v T},
4653     t = {\c t,\v t},
4654     U = {\`U,\^U,\^U,\"U,\H U,\r U},
4655     u = {\`u,\^u,\^u,\"u,\H u,\r u},
4656     Y = {\`Y,\"Y},
4657     y = {\`y,\"y},
4658     Z = {\`Z,\.Z,\v Z},
4659     z = {\`z,\.z,\v z}

```

The ‘soft hyphen’ often has reduced right side bearing so that it may already be protruded, hence no inheritance.

```

4660 %   - = {127},
4661   }
4662

```

15.5.3 LY1

More characters: 008 (‘fl’), 012 (‘fi’), 014 (‘ffi’), 015 (‘ffl’), Æ, æ, Œ, œ.

```

4663 \DeclareCharacterInheritance
4664   { encoding = LY1 }
4665   { A = {\`A,\^A,\^A,\~A,\"A,\r A},
4666     a = {\`a,\^a,\^a,\~a,\"a,\r a},
4667     C = {\c C},
4668     c = {\c c},
4669     D = {\DH},
4670     E = {\`E,\^E,\^E,\"E},
4671     e = {\`e,\^e,\^e,\"e},
4672     f = {011}, % ff
4673     I = {\`I,\^I,\^I,\"I},

```

```

4674     i = {\~i,\'i,\^i,\"i,\i},
4675     L = {\L},
4676     l = {\l},
4677     N = {\~N},
4678     n = {\~n},
4679     O = {\~O,\'O,\^O,\~O,\"O,\O},
4680     o = {\~o,\'o,\^o,\~o,\"o,\o},
4681     S = {\v S},
4682     s = {\v s},
4683     U = {\~U,\'U,\^U,\"U},
4684     u = {\~u,\'u,\^u,\"u},
4685     Y = {\'Y,\"Y},
4686     y = {\'y,\"y},
4687     Z = {\v Z},
4688     z = {\v z}
4689 }
4690

```

15.5.4 OT4

The Polish OT1 extension. More interesting characters here: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

4691 \DeclareCharacterInheritance
4692 { encoding = OT4 }
4693 { A = {\k A},
4694   a = {\k a},
4695   C = {\'C},
4696   c = {\'c},
4697   E = {\k E},
4698   e = {\k e},
4699   f = {011}, % ff
4700   i = {\i},
4701   j = {\j},
4702   L = {\L},
4703   l = {\l},
4704   N = {\~N},
4705   n = {\'n},
4706   O = {\O,\'O},
4707   o = {\o,\'o},
4708   S = {\'S},
4709   s = {\'s},
4710   Z = {\'Z,\.Z},
4711   z = {\'z,\.z}
4712 }
4713

```

15.5.5 QX

The Central European QX encoding.¹⁷ Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

4714 \DeclareCharacterInheritance
4715 { encoding = QX }
4716 { A = {\~A,\'A,\^A,\~A,\"A,\k A,\AA},
4717   a = {\~a,\'a,\^a,\~a,\"a,\k a,\aa},
4718   C = {\'C,\c C},
4719   c = {\'c,\c c},
4720   D = {\DH},
4721   E = {\~E,\'E,\^E,\"E,\k E},
4722   e = {\~e,\'e,\^e,\"e,\k e},
4723   f = {011}, % ff

```

17 Contributed by Maciej Eder.

```

4724 I = {\`I,\`I,\^I,\^I,\k I},
4725 i = {\`i,\`i,\^i,\^i,\k i,\i},
4726 j = {\j},
4727 L = {\L},
4728 l = {\l},
4729 N = {\`N,\-N},
4730 n = {\`n,\-n},
4731 O = {\0,\`0,\^0,\^0,\-0,\^0},
4732 o = {\0,\`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.

```

4733 S = {\`S,\c S,\textcommabelow S,\v S},
4734 s = {\`s,\c s,\textcommabelow s,\v s},
4735 T = {\c T,\textcommabelow T},
4736 t = {\c t,\textcommabelow t},
4737 U = {\`U,\`U,\^U,\^U,\k U},
4738 u = {\`u,\`u,\^u,\^u,\k u},
4739 Y = {\`Y,\^Y},
4740 y = {\`y,\^y},
4741 Z = {\`Z,\^Z,\v Z},
4742 z = {\`z,\^z,\v z},
4743 . = \textellipsis
4744 }
4745

```

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.

```

4746 \DeclareCharacterInheritance
4747 { encoding = T5 }
4748 { A = {\`A,\`A,\-A,\h A,\d A,\^A,\u A,
4749       \`Acircumflex,\`Acircumflex,\-Acircumflex,\hAcircumflex,\dAcircumflex,
4750       \`Abreve,\`Abreve,\-Abreve,\hAbreve,\dAbreve},
4751   a = {\`a,\`a,\-a,\h a,\d a,\^a,\u a,
4752       \`acircumflex,\`acircumflex,\-acircumflex,\hacircumflex,\dacircumflex,
4753       \`abreve,\`abreve,\-abreve,\h\abreve,\d\abreve},
4754   D = {\DJ},
4755   d = {\dj},
4756   E = {\`E,\`E,\-E,\h E,\d E,\^E,
4757       \`Ecircumflex,\`Ecircumflex,\-Ecircumflex,\hEcircumflex,\dEcircumflex},
4758   e = {\`e,\`e,\-e,\h e,\d e,\^e,
4759       \`ecircumflex,\`ecircumflex,\-ecircumflex,\h\ecircumflex,\d\ecircumflex},
4760   I = {\`I,\`I,\-I,\h I,\d I},
4761   i = {\`i,\`i,\-i,\h i,\d i,\i},
4762   O = {\`O,\`O,\-O,\h O,\d O,\^O,\horn O,
4763       \`Ocircumflex,\`Ocircumflex,\-Ocircumflex,\hOcircumflex,\dOcircumflex,
4764       \`Ohorn,\`Ohorn,\-Ohorn,\hOhorn,\dOhorn},
4765   o = {\`o,\`o,\-o,\h o,\d o,\^o,\horn o,
4766       \`ocircumflex,\`ocircumflex,\-ocircumflex,\h\ocircumflex,\d\ocircumflex,
4767       \`ohorn,\`ohorn,\-ohorn,\h\ohorn,\d\ohorn},
4768   U = {\`U,\`U,\-U,\h U,\d U,\horn U,
4769       \`Uhorn,\`Uhorn,\-Uhorn,\hUhorn,\dUhorn},
4770   u = {\`u,\`u,\-u,\h u,\d u,\horn u,
4771       \`uhorn,\`uhorn,\-uhorn,\h\uhorn,\d\uhorn},
4772   Y = {\`Y,\`Y,\-Y,\h Y,\d Y},
4773   y = {\`y,\`y,\-y,\h y,\d y}
4774 }
4775

```

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.

```

4776 \DeclareCharacterInheritance
4777   { encoding = {EU1,EU2} }
4778   { A = {\`A,\`A,\^A,\-A,\"A,\r A,\k A,\u A},
4779     a = {\`a,\`a,\^a,\-a,\"a,\r a,\k a,\u a},
4780     C = {\`C,\c C,\v C},
4781     c = {\`c,\c c,\v c},
4782     D = {\v D,\DH},
4783     d = {\v d,\dj},
4784     E = {\`E,\`E,\^E,\"E,\k E,\v E},
4785     e = {\`e,\`e,\^e,\"e,\k e,\v e},
4786 %   f = {/f_f}, % sometimes /f_f, sometimes /ff
4787     G = {\u G},
4788     g = {\u g},
4789     I = {\`I,\`I,\^I,\"I,\.I},
4790     i = {\`i,\`i,\^i,\"i,\.i},
4791 %   j = {\j},
4792     L = {\L,\`L,\v L},
4793     l = {\l,\`l,\v l},
4794     N = {\`N,\-N,\v N},
4795     n = {\`n,\-n,\v n},
4796     O = {\0,\`0,\^0,\-0,\"0,\H 0},
4797     o = {\0,\`o,\^o,\-o,\"o,\H o},
4798     R = {\`R,\v R},
4799     r = {\`r,\v r},
4800     S = {\`S,\c S,\v S}, % \SS
4801     s = {\`s,\c s,\v s},
4802     T = {\c T,\v T},
4803     t = {\c t,\v t},
4804     U = {\`U,\`U,\^U,\"U,\H U,\r U},
4805     u = {\`u,\`u,\^u,\"u,\H u,\r u},
4806     Y = {\`Y,\"Y},
4807     y = {\`y,\"y},
4808     Z = {\`Z,\.Z,\v Z},
4809     z = {\`z,\.z,\v z}
4810   }
4811
4812 /m-t

```

15.5.8 Euro symbols

Make Euro symbols settings simpler.

```

4813 *zpeu
4814 \DeclareCharacterInheritance
4815   { encoding = U,
4816     family   = {zpeu,zpeus,eurosans} }
4817   { E = 128 }
4818
4819 /zpeu
4820 *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.

```

4821 \DeclareCharacterInheritance
4822   { encoding = {OT1,U},
4823     family   = mvs }

```

```

4824 { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
4825
4826 </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.

```

4827 <*m-t>
4828 %%% -----
4829 %%% TRACKING/LETTERSPACING
4830
4831 \SetTracking
4832 [ name = default,
4833   no ligatures = {f} ]
4834 { encoding = {OT1,T1,T2A,LY1,OT4,QX,EU2} }
4835 { }
4836

```

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).

```

4837 %%% -----
4838 %%% EXPANSION
4839
4840 \SetExpansion
4841 [ name = default ]
4842 { encoding = {OT1,OT4,QX,T1,LY1} }
4843 {
4844   A = 500,      a = 700,
4845   \AE = 500,    \ae = 700,
4846   B = 700,      b = 700,
4847   C = 700,      c = 700,
4848   D = 500,      d = 700,
4849   E = 700,      e = 700,
4850   F = 700,
4851   G = 500,      g = 700,
4852   H = 700,      h = 700,
4853   K = 700,      k = 700,
4854   M = 700,      m = 700,
4855   N = 700,      n = 700,
4856   O = 500,      o = 700,
4857   \OE = 500,    \oe = 700,
4858   P = 700,      p = 700,
4859   Q = 500,      q = 700,
4860   R = 700,
4861   S = 700,      s = 700,
4862   U = 700,      u = 700,
4863   W = 700,      w = 700,
4864   Z = 700,      z = 700,
4865   2 = 700,
4866   3 = 700,
4867   6 = 700,
4868   8 = 700,
4869   9 = 700
4870 }
4871

```

Settings for Cyrillic T2A encoding.¹⁹

19 Contributed by *Karl Karlsson*.

```

4872 \SetExpansion
4873   [ name      = T2A ]
4874   { encoding = T2A }
4875   {
4876     A = 500,      a = 700,
4877     B = 700,      b = 700,
4878     C = 700,      c = 700,
4879     D = 500,      d = 700,
4880     E = 700,      e = 700,
4881     F = 700,
4882     G = 500,      g = 700,
4883     H = 700,      h = 700,
4884     K = 700,      k = 700,
4885     M = 700,      m = 700,
4886     N = 700,      n = 700,
4887     O = 500,      o = 700,
4888     P = 700,      p = 700,
4889     Q = 500,      q = 700,
4890     R = 700,
4891     S = 700,      s = 700,
4892     U = 700,      u = 700,
4893     W = 700,      w = 700,
4894     Z = 700,      z = 700,
4895     2 = 700,
4896     3 = 700,
4897     6 = 700,
4898     8 = 700,
4899     9 = 700,
4900     \CYRA = 500,   \cyra = 700,
4901     \CYRB = 700,   \cyrb = 700,
4902     \CYRV = 700,   \cyrv = 700,
4903     \CYRG = 700,   \cyrg = 700,
4904     \CYRD = 700,   \cyrd = 700,
4905     \CYRE = 700,   \cyre = 700,
4906     \CYRZH = 700,  \cyrzh = 700,
4907     \CYRZ = 700,   \cyrz = 700,
4908     \CYRI = 700,   \cyri = 700,
4909     \CYRISHRT = 700, \cyrishrt = 700,
4910     \CYRK = 700,   \cyrk = 700,
4911     \CYRL = 700,   \cyrl = 700,
4912     \CYRM = 700,   \cyrm = 700,
4913     \CYRN = 700,   \cyrn = 700,
4914     \CYRO = 500,   \cyro = 700,
4915     \CYRP = 700,   \cyrp = 700,
4916     \CYRR = 700,   \cyrr = 700,
4917     \CYRS = 700,   \cyrs = 700,
4918     \CYRT = 700,   \cyrt = 700,
4919     \CYRU = 700,   \cyru = 700,
4920     \CYRF = 700,   \cyrf = 700,
4921     \CYRH = 700,   \cyrh = 700,
4922     \CYRC = 700,   \cyrc = 700,
4923     \CYRCH = 700,  \cyrch = 700,
4924     \CYRSH = 700,  \cyrsh = 700,
4925     \CYRSHCH = 700, \cyrshch = 700,
4926     \CYRHRSN = 700, \cyrhrdsn = 700,
4927     \CYRERY = 700, \cyrery = 700,
4928     \CYRSFTSN = 700, \cyrsftsn = 700,
4929     \CYREREV = 700, \cyrerev = 700,
4930     \CYRYU = 700,  \cyryu = 700,
4931     \CYRYA = 700,  \cyrya = 700
4932   }
4933

```

T5 encoding does not contain \AE, \ae, \OE and \oe.

```

4934 \SetExpansion

```

```

4935 [ name      = T5 ]
4936 { encoding = T5 }
4937 {
4938   A = 500,      a = 700,
4939   B = 700,      b = 700,
4940   C = 700,      c = 700,
4941   D = 500,      d = 700,
4942   E = 700,      e = 700,
4943   F = 700,
4944   G = 500,      g = 700,
4945   H = 700,      h = 700,
4946   K = 700,      k = 700,
4947   M = 700,      m = 700,
4948   N = 700,      n = 700,
4949   O = 500,      o = 700,
4950   P = 700,      p = 700,
4951   Q = 500,      q = 700,
4952   R = 700,
4953   S = 700,      s = 700,
4954   U = 700,      u = 700,
4955   W = 700,      w = 700,
4956   Z = 700,      z = 700,
4957   2 = 700,
4958   3 = 700,
4959   6 = 700,
4960   8 = 700,
4961   9 = 700
4962 }
4963
4964 </m-t>

```

15.8 Character protrusion

```

4965 %%% -----
4966 %%% PROTRUSION
4967

```

For future historians, Hàn Thế Thành's original settings (from `protcode.tex`, converted to microtype notation).

```

\SetProtrusion
[ name      = thanh ]
{ encoding = OT1 }
{
  A = {50,50},
  F = { ,50},
  J = {50, },
  K = { ,50},
  L = { ,50},
  T = {50,50},
  V = {50,50},
  W = {50,50},
  X = {50,50},
  Y = {50,50},
  k = { ,50},
  r = { ,50},
  t = { ,50},
  v = {50,50},
  w = {50,50},
  x = {50,50},
  y = {50,50},
  . = { ,700},    {,}= { ,700},
  : = { ,500},    ; = { ,500},
  ! = { ,200},    ? = { ,200},
  ( = {50, },    ) = { ,50},

```

```

- = { ,700},
\textendash      = { ,300},    \textemdash      = { ,200},
\textquoteleft   = {700, },    \textquoteright  = { ,700},
\textquotedblleft = {500, },    \textquotedblright = { ,500}
}

```

15.8.1 Normal

The default settings always use the most moderate value.

```

4968 <*cfg-t>
4969 \SetProtrusion
4970 <m-t> [ name      = default ]

```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```

4971 <bch> [ name      = bch-default ]

```

- Bitstream Letter Gothic (blg)

```

4972 <blg> [ name      = blg-default ]

```

- Computer Modern Roman (cmr)

```

4973 <cmr> [ name      = cmr-default ]

```

- Adobe Garamond (pad, padx, padj)

```

4974 <pad> [ name      = pad-default ]

```

- Minion²⁰ (pmnx, pmnj)

```

4975 <pmn> [ name      = pmnj-default ]

```

- Palatino (ppl, pplx, pplj)

```

4976 <ppl> [ name      = ppl-default ]

```

- Times (ptm, ptmx, ptmj)

```

4977 <ptm> [ name      = ptm-default ]

```

- URW Garamond (ugm)

```

4978 <ugm> [ name      = ugm-default ]
4979 <m-t|cmr|pmn> { }
4980 <bch|blg|pad|ugm> { encoding = OT1,
4981 <ppl|ptm> { encoding = {OT1,OT4},
4982 <bch>      family   = bch }
4983 <blg>      family   = blg }
4984 <pad>      family   = {pad,padx,padj }
4985 <ppl>      family   = {ppl,pplx,pplj }
4986 <ptm>      family   = {ptm,ptmx,ptmj }
4987 <ugm>      family   = ugm }
4988 {
4989 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm> A = {50,50},
4990 <ugm>      A = {50,100},
4991 <pad|ptm>  \AE = {50, },
4992 <ugm>      \AE = {150,50},
4993 <ugm>      B = { ,50},
4994 <bch|pad|pmn|ugm> C = {50, },
4995 <bch|pad|pmn> D = { ,50},
4996 <ugm>      D = { ,70},
4997 <ugm>      E = { ,50},

```



```

4998 <m-t|bch|cmr|pad|pmn|ptm>    F = { ,50},
4999 <ugm>      F = { ,70},
5000 <bch|pad|pmn>      G = {50, },
5001 <ugm>      G = {50,50},
5002 <blg>      I = {150,150},
5003 <m-t|cmr|pad|pmn|ppl|ptm|ugm>    J = {50, },
5004 <bch|blg>      J = {100, },
5005 <!blg>      K = { ,50},
5006 <blg>      K = {50, },
5007 <m-t|bch|cmr|pad|pmn|ppl>      L = { ,50},
5008 <blg>      L = { ,150},
5009 <ptm>      L = { ,80},
5010 <ugm>      L = { ,120},
5011 <bch|pad|pmn|ugm>      O = {50,50},
5012 <pad>      \OE = {50, },
5013 <ugm>      \OE = {50,50},
5014 <blg>      P = { ,100},
5015 <ugm>      P = { ,50},
5016 <bch|pad|pmn>      Q = {50,70},
5017 <ugm>      Q = {50,50},
5018 <bch>      R = { ,50},
5019 <ugm>      R = { ,70},
5020 <m-t|bch|cmr|pad|pmn|ppl|ptm>    T = {50,50},
5021 <blg>      T = {100,100},
5022 <ugm>      T = {70,70},
5023 <m-t|bch|cmr|pad|pmn|ppl|ptm>    V = {50,50},
5024 <blg|ugm>      V = {70,70},
5025 <m-t|bch|cmr|pad|pmn|ppl|ptm>    W = {50,50},
5026 <ugm>      W = {70,70},
5027 <m-t|bch|cmr|pad|pmn|ppl|ptm>    X = {50,50},
5028 <ugm>      X = {50,70},
5029 <m-t|bch|cmr|pad|pmn|ppl>      Y = {50,50},
5030 <blg|ptm|ugm>      Y = {80,80},
5031 <ugm>      Z = {50,50},
5032 <blg>      f = {150,100},
5033 <blg>      i = {150,150},
5034 <blg>      j = {100,100},
5035 <m-t|bch|cmr|pad|pmn|ppl|ptm>    k = { ,50},
5036 <ugm>      k = { ,70},
5037 <blg>      l = {150,150},
5038 <pmn>      l = { , -50},
5039 <pad|ppl>      p = {50,50},
5040 <ugm>      p = { ,50},
5041 <pad|ppl>      q = {50, },
5042 <!blg>      r = { ,50},
5043 <blg>      r = {100, 80},
5044 <cmr|pad|pmn>      t = { ,70},
5045 <bch>      t = { ,50},
5046 <blg>      t = {150, 80},
5047 <ugm>      t = { ,100},
5048 <m-t|bch|cmr|pad|pmn|ppl|ptm>    v = {50,50},
5049 <blg>      v = {100,100},
5050 <ugm>      v = {50,70},
5051 <m-t|bch|cmr|pad|pmn|ppl|ptm>    w = {50,50},
5052 <ugm>      w = {50,70},
5053 <!blg>      x = {50,50},
5054 <blg>      x = {100,100},
5055 <m-t|bch|pad|pmn>      y = { ,50},
5056 <blg>      y = { 50,100},
5057 <cmr|ppl|ptm>      y = {50,70},
5058 <ugm>      y = { ,70},

5059 <cmr>      O = { ,50},
5060 <m-t>      l = {50,50},
5061 <bch|blg|pad|ptm|ugm>      l = {150,150},
5062 <cmr>      l = {100,200},

```

```

5063 <pmn>      1 = { ,50},
5064 <ppl>      1 = {100,100},
5065 <bch|cmr|pad|ugm>      2 = {50,50},
5066 <blg>      2 = { ,100},
5067 <bch|pmn>      3 = {50, },
5068 <cmr|pad|ugm>      3 = {50,50},
5069 <blg>      3 = {100, },
5070 <m-t|pad>      4 = {50,50},
5071 <bch>      4 = {100,50},
5072 <blg>      4 = {100, },
5073 <cmr|ugm>      4 = {70,70},
5074 <pmn>      4 = {50, },
5075 <ptm>      4 = {70, },
5076 <cmr>      5 = { ,50},
5077 <pad>      5 = {50,50},
5078 <bch>      6 = {50, },
5079 <cmr>      6 = { ,50},
5080 <pad>      6 = {50,50},
5081 <m-t>      7 = {50,50},
5082 <bch|pad|pmn|ugm>      7 = {50,80},
5083 <blg>      7 = {100,100},
5084 <cmr|ptm>      7 = {50,100},
5085 <ppl>      7 = { ,50},
5086 <cmr>      8 = { ,50},
5087 <bch|pad>      9 = {50,50},
5088 <cmr>      9 = { ,50},
5089 <m-t|cmr|pad|pmn|ppl|ptm|ugm>      . = { ,700},
5090 <bch>      . = { ,600},
5091 <blg>      . = {400,500},
5092 <!blg>      {,}= { ,500},
5093 <blg>      {,}= {300,400},
5094 <m-t|cmr|pad|pmn|ppl|ptm|ugm>      : = { ,500},
5095 <bch>      : = { ,400},
5096 <blg>      : = {300,400},
5097 <m-t|bch|pad|pmn|ptm>      ; = { ,300},
5098 <blg>      ; = {200,300},
5099 <cmr|ppl>      ; = { ,500},
5100 <ugm>      ; = { ,400},
5101 <!blg>      ! = { ,100},
5102 <blg>      ! = {200,200},
5103 <m-t|pad|pmn|ptm>      ? = { ,100},
5104 <bch|cmr|ppl|ugm>      ? = { ,200},
5105 <blg>      ? = {150,150},
5106 <pmn>      " = {300,300},
5107 <m-t|bch|cmr|pad|pmn|ppl>      @ = {50,50},
5108 <ptm>      @ = {100,100},
5109 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm>      ~ = {200,250},
5110 <ugm>      ~ = {300,350},
5111 <pad|ppl|ptm>      & = {50,100},
5112 <ugm>      & = { ,100},
5113 <m-t|cmr|pad|pmn>      \% = {50,50},
5114 <bch>      \% = { ,50},
5115 <ppl|ptm>      \% = {100,100},
5116 <ugm>      \% = {50,100},
5117 <blg>      \# = {100,100},
5118 <m-t|ppl|ptm|ugm>      * = {200,200},
5119 <bch|pmn>      * = {200,300},
5120 <blg>      * = {150,200},
5121 <cmr|pad>      * = {300,300},
5122 <m-t|cmr|ppl|ptm>      + = {250,250},
5123 <bch>      + = {150,250},
5124 <pad>      + = {300,300},
5125 <blg|pmn>      + = {150,200},
5126 <ugm>      + = {250,300},
5127 <blg|ugm>      {=}= {200,200},

```

```

5128 <m-t|pad|pmn|ptm> ( = {100, }, ) = { ,200},
5129 <bch|ugm> ( = {200, }, ) = { ,200},
5130 <cmr|blg> ( = {300, }, ) = { ,300},
5131 <ppl> ( = {100, }, ) = { ,300},
5132 <bch|pmn> [ = {100, }, ] = { ,100},
5133 <blg> [ = {300,100}, ] = { ,300},

5134 <m-t|pad|pmn|ptm> / = {100,200},
5135 <bch> / = { ,200},
5136 <blg> / = {300,300},
5137 <cmr|ppl> / = {200,300},
5138 <ugm> / = {100,300},
5139 <m-t|ptm> - = {500,500},
5140 <bch|cmr|ppl> - = {400,500},
5141 <blg> - = {300,400},
5142 <pad> - = {300,500},
5143 <pmn> - = {200,400},
5144 <ugm> - = {500,600},
5145 <blg> <= {200,100}, >= {100,200},
5146 <blg> - = {150,250},
5147 <blg> | = {250,250},
5148 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
5149 <bch> \textendash = {200,300}, \textendash = {150,250},
5150 <cmr> \textendash = {400,300}, \textendash = {300,200},
5151 <pad|ppl|ptm> \textendash = {300,300}, \textendash = {200,200},
5152 <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).

```

5153 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
5154 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
5155 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
5156 <pad|ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
5157 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
5158 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
5159 <m-t|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
5160 <blg> \textquotedblright = {300,400}
5161 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
5162 <pad|ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5163 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
5164 }
5165

```

Greek uppercase letters are in OT1 encoding only.

```

5166 <*m-t|cmr|pmn>
5167 \SetProtrusion
5168 <m-t> [ name = OT1-default,
5169 <cmr> [ name = cmr-OT1,
5170 <pmn> [ name = pmnj-OT1,
5171 <m-t> load = default ]
5172 <cmr> load = cmr-default ]
5173 <pmn> load = pmnj-default ]
5174 <m-t> { encoding = OT1 }
5175 <cmr> { encoding = {OT1,OT4},
5176 <pmn> { encoding = OT1,
5177 <cmr> family = cmr }
5178 <pmn> family = pmnj }
5179 {
5180 <m-t|cmr> \AE = {50, },
5181 <pmn> \OE = {50, }
5182 <*cmr>
5183 "00 = { ,150}, % \Gamma
5184 "01 = {100,100}, % \Delta
5185 "02 = { 50, 50}, % \Theta
5186 "03 = {100,100}, % \Lambda

```

```

5187 "06 = { 50, 50}, % \Sigma
5188 "07 = {100,100}, % \Upsilon
5189 "08 = { 50, 50}, % \Phi
5190 "09 = { 50, 50} % \Psi

```

Remaining slots can be found in the source file.

```

5191 </cmr>
5192 }
5193
5194 </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).

```

5195 \SetProtrusion
5196 <m-t> [ name = T1-default,
5197 <bch> [ name = bch-T1,
5198 <blg> [ name = blg-T1,
5199 <cmr> [ name = cmr-T1,
5200 <pad> [ name = pad-T1,
5201 <pmn> [ name = pmn-T1,
5202 <ppl> [ name = ppl-T1,
5203 <ptm> [ name = ptm-T1,
5204 <ugm> [ name = ugm-T1,
5205 <m-t> load = default ]
5206 <bch> load = bch-default ]
5207 <blg> load = blg-default ]
5208 <cmr> load = cmr-default ]
5209 <pad> load = pad-default ]
5210 <pmn> load = pmn-default ]
5211 <ppl> load = ppl-default ]
5212 <ptm> load = ptm-default ]
5213 <ugm> load = ugm-default ]
5214 <m-t> { encoding = {T1,LY1,EU1,EU2} }
5215 <bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
5216 <blg|ptm|ugm> { encoding = {T1},
5217 <bch> family = bch }
5218 <blg> family = blg }
5219 <cmr> family = cmr }
5220 <pad> family = {pad,padx,padj} }
5221 <pmn> family = pmn }
5222 <ppl> family = {ppl,pplx,pplj} }
5223 <ptm> family = {ptm,ptmx,ptmj} }
5224 <ugm> family = ugm }
5225 {
5226 <m-t|cmr> \AE = {50, },
5227 <bch|pmn> \OE = {50, },
5228 <pmn> \TH = { ,50},
5229 <blg> \v L = { ,250},
5230 <blg> \v d = { ,250},
5231 <blg> \v l = { ,250},
5232 <blg> \v t = { ,250},
5233 <blg> 127 = {300,400},
5234 <blg> 156 = {100, }, % IJ
5235 <blg> 188 = { 80, 80}, % ij
5236 <m-t|bch|pad|pmn|ppl|ptm> _ = {100,100},
5237 <cmr> _ = {200,200},
5238 <ugm> _ = {100,200},
5239 <m-t|pad|pmn|ptm> \textbackslash = {100,200},
5240 <bch> \textbackslash = {150,200},
5241 <blg> \textbackslash = {250,300},
5242 <cmr|ppl> \textbackslash = {200,300},
5243 <ugm> \textbackslash = {100,300},
5244 <ugm> \textbar = {200,200},
5245 <blg> \textendash = {300,300}, \textemdash = {150,150},

```

```

5246 <big> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
5247 <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.

```

5248 <m-t|cmr|pad|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5249 <big> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
5250 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5251 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5252 <big> \guilsinglleft = {300,500}, \guilsinglright = {300,500},
5253 <cmr|pad|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5254 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
5255 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5256 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5257 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
5258 <big|pad|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5259 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
5260 <m-t|bch|cmr|pad|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
5261 <big> \textexclamdown = {200, }, \textquestiondown = {100, },
5262 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
5263 <m-t|cmr|pad|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
5264 <bch|big|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
5265 <m-t|bch|cmr|pad|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
5266 <pmn> \textless = {100, }, \textgreater = { ,100},
5267 <pmn> \textvisiblespace = {100,100} % not in LY1
5268 }
5269

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

5270 <*cmr>
5271 \SetProtrusion
5272 [ name = lmr-T1,
5273   load = cmr-T1 ]
5274 { encoding = {T1,LY1},
5275   family = lmr }
5276
5277 \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5278 }
5279
5280 </cmr>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).²¹

```

5281 <m-t|cmr|pmn>
5282 \SetProtrusion
5283 <m-t> [ name = T2A-default,
5284 <cmr> [ name = cmr-T2A,
5285 <pmn> [ name = pmnj-T2A,
5286 <m-t> load = default ]
5287 <cmr> load = cmr-default ]
5288 <pmn> load = pmnj-default ]
5289 { encoding = T2A,
5290 <m-t> }
5291 <cmr> family = cmr }
5292 <pmn> family = pmnj }
5293 {
5294 \CYRA = {50,50},
5295 \CYRG = { ,50},
5296 \CYRK = { ,50},
5297 \CYRT = {50,50},
5298 \CYRH = {50,50},
5299 \CYRU = {50,50},

```

21 Contributed by Karl Karlsson.

```

5300 <pmn> \CYRS = {50, },
5301 <pmn> \CYRO = {50,50},
5302 \cyrk = { ,50},
5303 \cyrg = { ,50},
5304 \cyrh = {50,50},
5305 <m-t|pmn> \cyru = {50,50},
5306 <cmr> \cyru = {50,70},
5307 <m-t> - = {100,100},
5308 <cmr> - = {200,200},
5309 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,400},
5310 <cmr> \textbackslash = {200,300}, \quotedblbase = {400,400},
5311 <pmn> \textbackslash = {100,200}, \quotedblbase = {300,300},
5312 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},
5313 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5314 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5315 <pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
5316 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5317 <pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
5318 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
5319 <pmn> \textless = {100, }, \textgreater = { ,100}
5320 }
5321
5322 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).²² It also includes some glyphs otherwise in TS1.

```

5323 <*m-t|ptm>
5324 \SetProtrusion
5325 <m-t> [ name = QX-default,
5326 <ptm> [ name = ptm-QX,
5327 <m-t> load = default ]
5328 <ptm> load = ptm-default ]
5329 <m-t> { encoding = QX }
5330 <ptm> { encoding = QX,
5331 <ptm> family = {ptm,ptmx,ptmj} }
5332 {
5333 \AE = {50, },
5334 <ptm> * = {200,200},
5335 {=} = {100,100},
5336 \textunderscore = {100,100},
5337 \textbackslash = {100,200},
5338 \quotedblbase = {400,400},
5339 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5340 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5341 \textexclamdown = {100, }, \textquestiondown = {100, },
5342 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
5343 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
5344 \textless = {200,100}, \textgreater = {100,200},
5345 \textminus = {200,200}, \textdegree = {300,300},
5346 <m-t> \copyright = {100,100}, \textregistered = {100,100}
5347 <ptm> \copyright = {100,150}, \textregistered = {100,150},
5348 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
5349 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
5350 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
5351 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
5352 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
5353 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
5354 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
5355 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
5356 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
5357 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
5358 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
5359 <ptm> \textperthousand = { ,50}

```

22 Contributed by Maciej Eder.

```

5360 }
5361
5362 </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.

```

5363 <*cmr|bch>
5364 \SetProtrusion
5365 <cmr> [ name = cmr-T5,
5366 <cmr> load = cmr-default ]
5367 <bch> [ name = bch-T5,
5368 <bch> load = bch-default ]
5369 { encoding = T5,
5370 <cmr> family = cmr }
5371 <bch> family = bch }
5372 {
5373 <bch> _ = {100,100},
5374 <bch> \textbackslash = {150,200},
5375 <cmr> \textbackslash = {200,300},
5376 <cmr> \textquotedblleft = {200,600},
5377 <cmr> \textquotedbl = {300,300},
5378 <bch> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5379 <cmr> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5380 <bch> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5381 <cmr> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5382 <bch> \guillemotleft = {200,200}, \guillemotright = {150,300},
5383 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5384 <bch> \textbraceleft = {200, }, \textbraceright = { ,300},
5385 <cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5386 \textless = {200,100}, \textgreater = {100,200}
5387 }
5388
5389 </cmr|bch>

```

Minion with lining numbers.

```

5390 <*pmn>
5391 \SetProtrusion
5392 [ name = pmnx-OT1,
5393 load = pmnj-default ]
5394 { encoding = OT1,
5395 family = pmnx }
5396 {
5397 1 = {230,180}
5398 }
5399
5400 \SetProtrusion
5401 [ name = pmnx-T1,
5402 load = pmnj-T1 ]
5403 { encoding = {T1,LY1},
5404 family = pmnx }
5405 {
5406 1 = {230,180}
5407 }
5408
5409 \SetProtrusion
5410 [ name = pmnx-T2A,
5411 load = pmnj-T2A ]
5412 { encoding = {T2A},
5413 family = pmnx }
5414 {
5415 1 = {230,180}
5416 }
5417
5418 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

5419 <*ptm>
5420 \SetProtrusion
5421 [ name = ptm-LY1,
5422   load = ptm-T1 ]
5423 { encoding = LY1,
5424   family = {ptm,ptmx,ptmj} }
5425 {
5426   - = {100,100},
5427   \texttrademark = {100,100},
5428   \textregistered = {100,100},
5429   \textcopyright = {100,100},
5430   \textdegree = {300,300},
5431   \textminus = {200,200},
5432   \textellipsis = {150,200},
5433 % \texteuro = { , }, % ?
5434   \textcent = {100,100},
5435   \textquotesingle = {500,500},
5436   \textflorin = { 50, 70},
5437   \textdagger = {150,150},
5438   \textdaggerdbl = {100,100},
5439   \textperthousand = { , 50},
5440   \textbullet = {150,150},
5441   \textonesuperior = {100,100},
5442   \texttwosuperior = { 50, 50},
5443   \textthreesuperior = { 50, 50},
5444   \textperiodcentered = {300,300},
5445   \textplusminus = { 50, 80},
5446   \textmultiply = {100,100},
5447   \textdivide = { 50,150}

```

Remaining slots in the source file.

```

5448 }
5449
5450 </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

```

5451 \SetProtrusion
5452 <m-t> [ name = OT1-it ]
5453 <bch> [ name = bch-it ]
5454 <blg> [ name = blg-it,
5455   load = blg-default ]
5456 <cmr> [ name = cmr-it ]
5457 <pad> [ name = pad-it ]
5458 <pmn> [ name = pmn-it ]
5459 <ppl> [ name = ppl-it ]
5460 <ptm> [ name = ptm-it ]
5461 <ugm> [ name = ugm-it ]
5462 <m-t|bch|blg|pad|ugm> { encoding = OT1,
5463 <ppl|ptm> { encoding = {OT1,OT4},
5464 <bch> family = bch,
5465 <blg> family = blg,

```

23 Settings contributed by Hendrik Vogt.


```

5466 <pad>      family = {pad,padx,padj},
5467 <ppl>      family = {ppl,pplx,pplj},
5468 <ptm>      family = {ptm,ptmx,ptmj},
5469 <ugm>      family = ugm,
5470 <m-t|bch|pad|ppl|ptm> shape = {it,sl} }
5471 <blg|ugm>   shape = it }
5472 <cmr|pmn>   { }
5473 {
5474 <cmr>      A = {100,100},
5475 <ptm>      A = {100,50},
5476 <pad|pmn>   A = {50, },
5477 <ugm>      A = { ,150},
5478 <ppl>      A = {50,50},
5479 <ptm>      \AE = {100, },
5480 <pad|ppl>   \AE = {50, },
5481 <cmr>      B = {83,-40},
5482 <pad|ppl|ptm> B = {50, },
5483 <pmn>      B = {20,-50},
5484 <bch|ppl|ptm|ugm> C = {50, },
5485 <cmr>      C = {165,-75},
5486 <pad>      C = {100, },
5487 <pmn>      C = {50,-50},
5488 <cmr>      D = {75, -28},
5489 <pad|ppl|ptm> D = {50,50},
5490 <pmn>      D = {20, },
5491 <cmr>      E = {80,-55},
5492 <pad|ppl|ptm> E = {50, },
5493 <pmn>      E = {20,-50},
5494 <cmr>      F = {85,-80},
5495 <pad|ptm>   F = {100, },
5496 <pmn>      F = {10, },
5497 <ppl>      F = {50, },
5498 <bch|ppl|ptm|ugm> G = {50, },
5499 <cmr>      G = {153,-15},
5500 <pad>      G = {100, },
5501 <pmn>      G = {50,-50},
5502 <cmr>      H = {73,-60},
5503 <pad|ppl|ptm> H = {50, },
5504 <cmr>      I = {140,-120},
5505 <pad|ptm>   I = {50, },
5506 <pmn>      I = {20,-50},
5507 <cmr>      J = {135,-80},
5508 <pad>      J = {50, },
5509 <pmn>      J = {20, },
5510 <ptm>      J = {100, },
5511 <cmr>      K = {70,-30},
5512 <pad|ppl|ptm> K = {50, },
5513 <pmn>      K = {20, },
5514 <cmr>      L = {87, 40},
5515 <pad|ppl|ptm> L = {50, },
5516 <pmn>      L = {20,50},
5517 <ugm>      L = { ,100},
5518 <cmr>      M = {67,-45},
5519 <pmn>      M = { , -30},
5520 <ptm>      M = {50, },
5521 <cmr>      N = {75,-55},
5522 <pmn>      N = { , -30},
5523 <ptm>      N = {50, },
5524 <bch|pmn|ppl|ptm> O = {50, },
5525 <cmr>      O = {150,-30},
5526 <pad>      O = {100, },
5527 <ugm>      O = {70,50},
5528 <ppl|ptm>   \OE = {50, },
5529 <pad>      \OE = {100, },
5530 <cmr>      P = {82,-50},

```

```

5531 <pad|ppl|ptm>      P = {50, },
5532 <pmn>              P = {20,-50},
5533 <bch|pmn|ppl|ptm>   Q = {50, },
5534 <cmr>              Q = {150,-30},
5535 <pad>              Q = {100, },
5536 <ugm>              Q = {70,50},
5537 <cmr>              R = {75, 15},
5538 <pad|ppl|ptm>      R = {50, },
5539 <pmn>              R = {20, },
5540 <bch|pad|ppl|ptm>   S = {50, },
5541 <cmr>              S = {90,-65},
5542 <pmn>              S = {20,-30},
5543 <bch|pad|ppl|ptm>   $ = {50, },
5544 <cmr>              $ = {100,-20},
5545 <pmn>              $ = {20,-30},
5546 <bch|pmn|ugm>      T = {70, },
5547 <cmr>              T = {220,-85},
5548 <pad|ppl|ptm>      T = {100, },
5549 <cmr>              U = {230,-55},
5550 <pad|ppl|ptm>      U = {50, },
5551 <pmn>              U = {50,-50},
5552 <cmr>              V = {260,-60},
5553 <pad|pmn|ugm>      V = {100, },
5554 <ppl|ptm>          V = {100,50},
5555 <cmr>              W = {185,-55},
5556 <pad|pmn|ugm>      W = {100, },
5557 <ppl>              W = {50, },
5558 <ptm>              W = {100,50},
5559 <cmr>              X = {70,-30},
5560 <ppl|ptm>          X = {50, },
5561 <cmr>              Y = {250,-60},
5562 <pmn>              Y = {50, },
5563 <ppl>              Y = {100,50},
5564 <ptm>              Y = {100, },
5565 <cmr>              Z = {90,-60},
5566 <pmn>              Z = { , -50},
5567 <cmr>              a = {150,-10},
5568 <cmr>              b = {170, },
5569 <cmr>              c = {173,-10},
5570 <cmr>              d = {150,-55},
5571 <pmn>              d = { , -50},
5572 <cmr>              e = {180, },
5573 <cmr>              f = { , -250},
5574 <pad|pmn>          f = { , -100},
5575 <cmr>              g = {150,-10},
5576 <cmr>              h = {100, },
5577 <cmr>              i = {210, },
5578 <pmn>              i = { , -30},
5579 <cmr>              j = { , -40},
5580 <pmn>              j = { , -30},
5581 <cmr>              k = {110,-50},
5582 <cmr>              l = {240,-110},
5583 <pmn>              l = { , -100},
5584 <cmr>              m = {80, },
5585 <cmr>              n = {115, },
5586 <bch>              o = {50,50},
5587 <cmr>              o = {155, },
5588 <bch>              p = { , 50},
5589 <pmn>              p = {-50, },
5590 <bch>              q = {50, },
5591 <cmr>              q = {170,-40},
5592 <cmr>              r = {155,-40},
5593 <pmn>              r = { , 50},
5594 <cmr>              s = {130, },
5595 <bch>              t = { , 50},

```

```

5596 <cmr>      t = {230,-10},
5597 <cmr>      u = {120, },
5598 <cmr>      v = {140,-25},
5599 <pmn|ugm>   v = {50, },
5600 <bch>      w = { ,50},
5601 <cmr>      w = {98,-20},
5602 <pmn|ugm>   w = {50, },
5603 <cmr>      x = {65,-40},
5604 <bch>      y = { ,50},
5605 <cmr>      y = {130,-20},
5606 <cmr>      z = {110,-80},
5607 <cmr>      0 = {170,-85},
5608 <bch|ptm>   1 = {150,100},
5609 <cmr>      1 = {230,110},
5610 <pad>      1 = {150, },
5611 <pmn>      1 = {50, },
5612 <ppl>      1 = {100, },
5613 <ugm>      1 = {150,150},
5614 <cmr>      2 = {130,-70},
5615 <pad|ppl|ptm> 2 = {50, },
5616 <pmn>      2 = {-50, },
5617 <bch>      3 = {50, },
5618 <cmr>      3 = {140,-70},
5619 <pmn>      3 = {-100, },
5620 <ptm>      3 = {100,50},
5621 <bch>      4 = {100, },
5622 <cmr>      4 = {130,80},
5623 <pad>      4 = {150, },
5624 <ppl|ptm>   4 = {50, },
5625 <cmr>      5 = {160, },
5626 <ptm>      5 = {50, },
5627 <bch>      6 = {50, },
5628 <cmr>      6 = {175,-30},
5629 <bch|pad|ptm> 7 = {100, },
5630 <cmr>      7 = {250,-150},
5631 <pmn>      7 = {20, },
5632 <ppl>      7 = {50, },
5633 <cmr>      8 = {130,-40},
5634 <cmr>      9 = {155,-80},
5635 <m-t|cmr|pad|pmn|ppl> . = { ,500},
5636 <blg>      . = {400,600},
5637 <bch|ptm|ugm> . = { ,700},
5638 <blg>      {,}= {300,500},
5639 <m-t|pad|pmn|ppl> {,}= { ,500},
5640 <cmr>      {,}= { ,450},
5641 <bch|ugm>   {,}= { ,600},
5642 <ptm>      {,}= { ,700},
5643 <m-t|cmr|pad|ppl> : = { ,300},
5644 <bch|ugm>   : = { ,400},
5645 <pmn>      : = { ,200},
5646 <ptm>      : = { ,500},
5647 <m-t|cmr|pad|ppl> ; = { ,300},
5648 <bch|ugm>   ; = { ,400},
5649 <pmn>      ; = { ,200},
5650 <ptm>      ; = { ,500},
5651 <ptm>      ! = { ,100},
5652 <bch>      ? = { ,200},
5653 <ptm>      ? = { ,100},
5654 <ppl>      ? = { ,300},
5655 <pmn>      " = {400,200},
5656 <m-t|pad|pmn|ppl|ptm> & = {50,50},
5657 <bch>      & = { ,80},
5658 <cmr>      & = {130,30},
5659 <ugm>      & = {50,100},
5660 <m-t|pad|pmn> \% = {100, },

```

```

5661 <cmr> \% = {180,50},
5662 <bch> \% = {50,50},
5663 <ppl|ptm> \% = {100,100},
5664 <ugm> \% = {100,50},
5665 <m-t|pmn|ppl> * = {200,200},
5666 <bch> * = {300,200},
5667 <cmr> * = {380,20},
5668 <pad> * = {500,100},
5669 <ptm|ugm> * = {400,200},
5670 <m-t|pmn|ppl> + = {150,200},
5671 <cmr> + = {180,200},
5672 <bch|ugm> + = {250,250},
5673 <pad|ptm> + = {250,200},
5674 <m-t|pad|pmn|ppl> @ = {50,50},
5675 <bch> @ = {80,50},
5676 <cmr> @ = {180,10},
5677 <ptm> @ = {150,150},
5678 <m-t|bch|ugm> ~ = {150,150},
5679 <cmr|pad|pmn|ppl|ptm> ~ = {200,150},
5680 <ugm> {=} = {200,200},
5681 <m-t|bch|pad|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
5682 <cmr> ( = {300, }, ) = { ,70},
5683 <m-t|pad|ppl|ptm|ugm> / = {100,200},
5684 <cmr> / = {100,100},
5685 <bch> / = { ,150},
5686 <pmn> / = {100,150},
5687 <m-t> - = {300,300},
5688 <bch|pad> - = {300,400},
5689 <pmn> - = {200,300},
5690 <cmr> - = {500,300},
5691 <ppl> - = {300,500},
5692 <ptm> - = {500,500},
5693 <ugm> - = {400,700},
5694 <blg> - = {0,300},
5695 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
5696 <bch> \textendash = {200,300}, \textendash = {150,200},
5697 <cmr> \textendash = {500,300}, \textendash = {400,170},
5698 <pad|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
5699 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
5700 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
5701 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
5702 <pad> \textquoteleft = {800,200}, \textquoteright = {800,200},
5703 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
5704 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
5705 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
5706 <blg> \textquotedblright = {300,300}
5707 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
5708 <pad> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
5709 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
5710 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
5711 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
5712 }
5713
5714 <*cmr|pmn>
5715 \SetProtrusion
5716 <cmr> [ name = cmr-it-OT1,
5717 <pmn> [ name = pmnj-it-OT1,
5718 <cmr> load = cmr-it ]
5719 <pmn> load = pmnj-it ]
5720 <cmr> { encoding = {OT1,OT4},
5721 <pmn> { encoding = OT1,
5722 <cmr> family = cmr,
5723 <pmn> family = pmnj,
5724 <cmr> shape = it }
5725 <pmn> shape = {it,sl} }

```

```

5726 {
5727   <cmr>    \AE = {100, },
5728   <pmn>    \AE = { , -50},
5729   <cmr>    \OE = {100, },
5730   <pmn>    \OE = {50, }
5731   <*cmr>
5732   "00 = {200,150}, % \Gamma
5733   "01 = {150,100}, % \Delta
5734   "02 = {150, 50}, % \Theta
5735   "03 = {150, 50}, % \Lambda
5736   "04 = {100,100}, % \Xi
5737   "05 = {100,100}, % \Pi
5738   "06 = {100, 50}, % \Sigma
5739   "07 = {200,150}, % \Upsilon
5740   "08 = {150, 50}, % \Phi
5741   "09 = {150,100}, % \Psi
5742   "0A = { 50, 50} % \Omega
5743   </cmr>
5744 }
5745
5746 </cmr|pmn>
5747 \SetProtrusion
5748 <m-t> [ name = Tl-it-default,
5749 <bch> [ name = bch-it-Tl,
5750 <blg> [ name = blg-it-Tl,
5751 <cmr> [ name = cmr-it-Tl,
5752 <pad> [ name = pad-it-Tl,
5753 <pmn> [ name = pmnj-it-Tl,
5754 <ppl> [ name = ppl-it-Tl,
5755 <ptm> [ name = ptm-it-Tl,
5756 <ugm> [ name = ugm-it-Tl,
5757 <m-t> load = OTl-it ]
5758 <bch> load = bch-it ]
5759 <blg> load = blg-Tl ]
5760 <cmr> load = cmr-it ]
5761 <pmn> load = pmnj-it ]
5762 <pad> load = pad-it ]
5763 <ppl> load = ppl-it ]
5764 <ptm> load = ptm-it ]
5765 <ugm> load = ugm-it ]
5766 <m-t|bch|cmr|pad|pmn|ppl> { encoding = {Tl,LYl},
5767 <blg|ptm|ugm> { encoding = Tl,
5768 <bch> family = bch,
5769 <blg> family = blg,
5770 <cmr> family = cmr,
5771 <pmn> family = pmnj,
5772 <pad> family = {pad,padx,padj},
5773 <ppl> family = {ppl,pplx,pplj},
5774 <ptm> family = {ptm,ptmx,ptmj},
5775 <ugm> family = ugm,
5776 <m-t|bch|pad|pmn|ppl|ptm> shape = {it,sl} }
5777 <blg|cmr|ugm> shape = it }
5778 {
5779 <m-t|bch|pmn> _ = { ,100},
5780 <blg> _ = {0,300},
5781 <cmr|ugm> _ = {100,200},
5782 <pad|ppl|ptm> _ = {100,100},
5783 <blg> . = {400,600},
5784 <blg> {,}= {300,500},
5785 <cmr> \AE = {100, },
5786 <pmn> \AE = { , -50},
5787 <bch|pmn> \OE = { 50, },
5788 <cmr> \OE = {100, },
5789 <pmn> 031 = { , -100}, % ffl
5790 <cmr|ptm> 156 = {100, }, % IJ

```

```

5791 <pad>      156 = {50, }, % IJ
5792 <pmn>      156 = {20, }, % IJ
5793 <pmn>      188 = { , -30}, % ij
5794 <pmn>      \v t = { , 100},
5795 <m-t|pad|ppl|ptm> \textbackslash = {100,200},
5796 <cmr|ugm>      \textbackslash = {300,300},
5797 <bch>         \textbackslash = {150,150},
5798 <pmn>         \textbackslash = {100,150},
5799 <ugm>         \textbar = {200,200},
5800 <cmr>         \textquotedblleft = {500,300},
5801 <blg>         \textquoteleft = {400,400}, \textquoteright = {400,400},
5802 <blg>         \textquotedbl = {300,300}, \textquotedblleft = {300,300},
5803 <blg>         \textquotedblright = {300,300}, \quotedblbase = {200,600},
5804 <m-t|ptm>      \quotesinglbase = {300,700}, \quotedblbase = {400,500},
5805 <cmr>         \quotesinglbase = {300,700}, \quotedblbase = {200,600},
5806 <bch|pmn>      \quotesinglbase = {200,500}, \quotedblbase = {150,500},
5807 <pad|ppl>      \quotesinglbase = {500,500}, \quotedblbase = {400,400},
5808 <ugm>         \quotesinglbase = {300,700}, \quotedblbase = {300,500},
5809 <m-t|ppl|ptm>  \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5810 <bch|pmn>      \guilsinglleft = {300,400}, \guilsinglright = {200,500},
5811 <cmr>         \guilsinglleft = {500,300}, \guilsinglright = {400,400},
5812 <pad>         \guilsinglleft = {500,400}, \guilsinglright = {300,500},
5813 <ugm>         \guilsinglleft = {400,400}, \guilsinglright = {300,600},
5814 <m-t|ppl>      \guillemotleft = {300,300}, \guillemotright = {300,300},
5815 <bch|pmn>      \guillemotleft = {200,300}, \guillemotright = {150,400},
5816 <cmr>         \guillemotleft = {400,100}, \guillemotright = {200,300},
5817 <pad>         \guillemotleft = {300,300}, \guillemotright = {200,400},
5818 <ptm>         \guillemotleft = {300,400}, \guillemotright = {200,400},
5819 <ugm>         \guillemotleft = {300,400}, \guillemotright = {300,400},
5820 <m-t|pad|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
5821 <cmr|ptm>      \textexclamdown = {200, }, \textquestiondown = {200, },
5822 <pmn>         \textexclamdown = {-50, }, \textquestiondown = {-50, },
5823 <m-t|ppl|ugm>  \textbraceleft = {200,100}, \textbraceright = {200,200},
5824 <bch|pmn>      \textbraceleft = {200, }, \textbraceright = { , 200},
5825 <cmr|pad|ptm>  \textbraceleft = {400,100}, \textbraceright = {200,200},
5826 <bch|pmn>      \textless = {100, }, \textgreater = { , 100},
5827 <cmr|pad|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
5828 <pmn>         \textvisiblespace = {100,100}
5829 }
5830
5831 <*m-t|cmr|pmn>
5832 \SetProtrusion
5833 <m-t> [ name = T2A-it-default,
5834 <cmr> [ name = cmr-it-T2A,
5835 <pmn> [ name = pmnj-it-T2A,
5836 <m-t> load = OT1-it ]
5837 <cmr> load = cmr-it ]
5838 <pmn> load = pmnj-it ]
5839 { encoding = T2A,
5840 <cmr> family = cmr,
5841 <pmn> family = pmnj,
5842 <m-t|pmn> shape = {it,sl} }
5843 <cmr> shape = it }
5844 {
5845 <cmr> \CYRA = {100,50},
5846 <pmn> \CYRA = {50, },
5847 <cmr> \CYRB = {50, },
5848 <cmr> \CYRV = {50, },
5849 <pmn> \CYRV = {20,-50},
5850 <cmr> \CYRG = {100, },
5851 <pmn> \CYRG = {10, },
5852 <cmr> \CYRD = {50, },
5853 <cmr> \CYRE = {50, },
5854 <pmn> \CYRE = {20,-50},
5855 <cmr> \CYRZH = {50, },

```

```

5856 <cmr> \CYRZ = {50, },
5857 <pmn> \CYRZ = {20,-50},
5858 <cmr> \CYRI = {50, },
5859 <pmn> \CYRI = { , -30},
5860 <cmr> \CYRISHRT = {50, },
5861 <cmr> \CYRK = {50, },
5862 <pmn> \CYRK = {20, },
5863 <cmr> \CYRL = {50, },
5864 <cmr> \CYRM = {50, },
5865 <pmn> \CYRM = { , -30},
5866 <cmr> \CYRN = {50, },
5867 <cmr> \CYRO = {100, },
5868 <pmn> \CYRO = {50, },
5869 <cmr> \CYRP = {50, },
5870 <cmr> \CYRR = {50, },
5871 <pmn> \CYRR = {20,-50},
5872 <cmr> \CYRS = {100, },
5873 <pmn> \CYRS = {50, },
5874 <cmr> \CYRT = {100, },
5875 <pmn> \CYRT = {70, },
5876 <cmr> \CYRU = {100, },
5877 <pmn> \CYRU = {50, },
5878 <cmr> \CYRF = {100, },
5879 <cmr> \CYRH = {50, },
5880 <cmr> \CYRC = {50, },
5881 <cmr> \CYRCH = {100, },
5882 <cmr> \CYRSH = {50, },
5883 <cmr> \CYRSHCH = {50, },
5884 <cmr> \CYRHRDSN = {100, },
5885 <cmr> \CYRERY = {50, },
5886 <cmr> \CYRSFTSN = {50, },
5887 <cmr> \CYREREV = {50, },
5888 <cmr> \CYRYU = {50, },
5889 <cmr> \CYRYA = {50, },
5890 <pmn> \CYRYA = { , 20},
5891 <pmn> \cyrr = {-50, },
5892 <m-t|pmn> _ = { , 100},
5893 <cmr> _ = {100,200},
5894 <pmn> 031 = { , -100}, % ff1
5895 <pmn> \v t = { , 100},
5896 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
5897 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
5898 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
5899 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
5900 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
5901 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
5902 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
5903 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
5904 <pmn> \textbraceleft = {200, }, \textbraceright = { , 200},
5905 <cmr> \textquotedblleft = {500,300},
5906 <cmr> \textless = {300,100}, \textgreater = {200,100}
5907 <pmn> \textless = {100, }, \textgreater = { , 100}
5908 }
5909
5910 </m-t|cmr|pmn>
5911 <*m-t|ptm>
5912 \SetProtrusion
5913 <m-t> [ name = QX-it-default,
5914 <ptm> [ name = ptm-it-QX,
5915 <m-t> load = OT1-it ]
5916 <ptm> load = ptm-it ]
5917 { encoding = {QX},
5918 <ptm> family = {ptm,ptmx,ptmj},
5919 shape = {it,sl} }
5920 {

```

```

5921 <ptm>    009 = { , 50}, % fk
5922 {=} = {100,100},
5923 <m-t>    \textunderscore = {100,100},
5924 <ptm>    \textunderscore = {100,150},
5925    \textbackslash = {100,200},
5926    \quotedblbase = {300,400},
5927 <m-t>    \guillemotleft = {300,300}, \guillemotright = {300,300},
5928 <ptm>    \guillemotleft = {200,400}, \guillemotright = {200,400},
5929    \textexclamdown = {200, }, \textquestiondown = {200, },
5930    \textbraceleft = {200,100}, \textbraceright = {200,200},
5931    \textless = {100,100}, \textgreater = {100,100},
5932    \textminus = {200,200}, \textdegree = {300,150},
5933 <m-t>    \copyright = {100,100}, \textregistered = {100,100}
5934 <ptm>    \textregistered = {100,150}, \copyright = {100,150},
5935 <ptm>    \textDelta = { 70, }, \textdelta = { , 50},
5936 <ptm>    \textpi = { 50, 80}, \textmu = { , 80},
5937 <ptm>    \texteuro = {200, }, \textellipsis = {100,200},
5938 <ptm>    \textquoteleft = {500,400}, \textquoteright = {500,400},
5939 <ptm>    \textquotedblleft = {500,300}, \textquotedblright = {400,400},
5940 <ptm>    \textapprox = { 50, 50}, \textinfty = {100,100},
5941 <ptm>    \textdagger = {150,150}, \textdaggerdbl = {100,100},
5942 <ptm>    \textdiv = {150,150}, \textasciitilde = { 80, 80},
5943 <ptm>    \texttimes = {100,150}, \textpm = { 50, 80},
5944 <ptm>    \textbullet = {300,100}, \textperiodcentered = {300,300},
5945 <ptm>    \textquotesingle = {500,500}, \textquotedbl = {300,300},
5946 <ptm>    \textperthousand = { ,50}
5947 }
5948
5949 </m-t|ptm>
5950 <*cmr|bch>
5951 \SetProtrusion
5952 <cmr> [ name = cmr-it-T5,
5953 <cmr> load = cmr-it ]
5954 <bch> [ name = bch-it-T5,
5955 <bch> load = bch-it ]
5956 { encoding = T5,
5957 <bch> family = bch,
5958 <cmr> family = cmr,
5959 shape = it }
5960 {
5961 <bch> _ = { ,100},
5962 <cmr> _ = {100,200},
5963 <bch> \textbackslash = {150,150},
5964 <cmr> \textbackslash = {300,300},
5965 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
5966 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
5967 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
5968 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
5969 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
5970 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
5971 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
5972 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
5973 <bch> \textless = {100, }, \textgreater = { ,100}
5974 <cmr> \textless = {300,100}, \textgreater = {200,100}
5975 }
5976
5977 </cmr|bch>

```

Slanted is very similar to italic.

```

5978 <*cmr>
5979 \SetProtrusion
5980 [ name = cmr-sl,
5981 load = cmr-it-OT1 ]
5982 { encoding = {OT1,OT4},
5983 family = cmr,

```



```

5984     shape    = sl }
5985     {
5986         L = { ,50},
5987         f = { ,-50},
5988         - = {300, },
5989         \textendash = {400, }, \textemdash = {300, }
5990     }
5991
5992 \SetProtrusion
5993 [ name    = cmr-sl-T1,
5994   load    = cmr-it-T1 ]
5995 { encoding = {T1,LY1},
5996   family   = cmr,
5997   shape    = sl }
5998 {
5999     L = { ,50},
6000     f = { ,-50},
6001     - = {300, },
6002     \textendash = {400, }, \textemdash = {300, }
6003 }
6004
6005 \SetProtrusion
6006 [ name    = cmr-sl-T2A,
6007   load    = cmr-it-T2A ]
6008 { encoding = T2A,
6009   family   = cmr,
6010   shape    = sl }
6011 {
6012     L = { ,50},
6013     f = { ,-50},
6014     - = {300, },
6015     \textendash = {400, }, \textemdash = {300, }
6016 }
6017
6018 \SetProtrusion
6019 [ name    = cmr-sl-T5,
6020   load    = cmr-it-T5 ]
6021 { encoding = T5,
6022   family   = cmr,
6023   shape    = sl }
6024 {
6025     L = { ,50},
6026     f = { ,-50},
6027     - = {300, },
6028     \textendash = {400, }, \textemdash = {300, }
6029 }
6030
6031 \SetProtrusion
6032 [ name    = lmr-it-T1,
6033   load    = cmr-it-T1 ]
6034 { encoding = {T1,LY1},
6035   family   = lmr,
6036   shape    = {it,sl} }
6037 {
6038     \textquotedblleft = { ,200}, \textquotedblright = { ,200},
6039     \quotesinglbase    = { ,400}, \quotedblbase    = { ,500}
6040 }
6041

```

Oldstyle numerals are slightly different.

```

6042 \SetProtrusion
6043 [ name = cmr(oldstyle)-it,
6044   load = cmr-it-T1 ]
6045 { encoding = T1,
6046   family   = {hfor,cmor},

```

```

6047     shape    = {it,s1} }
6048   {
6049     1 = {250, 50},
6050     2 = {150,-100},
6051     3 = {100,-50},
6052     4 = {150,150},
6053     6 = {200,  },
6054     7 = {200, 50},
6055     8 = {150,-50},
6056     9 = {100, 50}
6057   }
6058
6059 </cmr>
6060 < *pmn>
6061 \SetProtrusion
6062   [ name    = pmnx-it,
6063     load    = pmnj-it ]
6064   { encoding = OT1,
6065     family   = pmnx,
6066     shape    = {it,s1} }
6067   {
6068     1 = {100,150}
6069   }
6070
6071 \SetProtrusion
6072   [ name    = pmnx-it-T1,
6073     load    = pmnj-it-T1 ]
6074   { encoding = {T1,LY1},
6075     family   = pmnx,
6076     shape    = {it,s1} }
6077   {
6078     1 = {100,150}
6079   }
6080
6081 \SetProtrusion
6082   [ name    = pmnx-it-T2A,
6083     load    = pmnj-it-T2A ]
6084   { encoding = {T2A},
6085     family   = pmnx,
6086     shape    = {it,s1} }
6087   {
6088     1 = {100,150}
6089   }
6090
6091 </pmn>
6092 < *ptm>
6093 \SetProtrusion
6094   [ name    = ptm-it-LY1,
6095     load    = ptm-it-T1 ]
6096   { encoding = {LY1},
6097     family   = {ptm,ptmx,ptmj},
6098     shape    = {it,s1} }
6099   {
6100     -                      = {100,100},
6101     \texttrademark        = {100,100},
6102     \textregistered       = {100,100},
6103     \textcopyright        = {100,100},
6104     \textdegree           = {300,100},
6105     \textminus            = {200,200},
6106     \textellipsis        = {100,200},
6107     \% \texteuro          = { ,  }, % ?
6108     \textcent             = {100,100},
6109     \textquotesingle      = {500,  },
6110     \textflorin           = {100, 70},
6111     \textdagger           = {150,150},

```

```

6112 \textdaggerdbl      = {100,100},
6113 \textbullet          = {150,150},
6114 \textonesuperior     = {150,100},
6115 \texttwosuperior     = {150, 50},
6116 \textthreesuperior   = {150, 50},
6117 \textparagraph      = {100,  },
6118 \textperiodcentered  = {500,300},
6119 \textonequarter      = { 50,  },
6120 \textonehalf         = { 50,  },
6121 \textplusminus       = {100,100},
6122 \textmultiply        = {150,150},
6123 \textdivide         = {150,150}
6124 }
6125
6126 </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).

```

6127 <*(blg|ugm)>
6128 \SetProtrusion
6129 <m-t> [ name      = OT1-sc,
6130 <bch> [ name      = bch-sc,
6131 <cmr> [ name      = cmr-sc-OT1,
6132 <pad> [ name      = pad-sc,
6133 <pmn> [ name      = pmnj-sc,
6134 <ppl> [ name      = ppl-sc,
6135 <ptm> [ name      = ptm-sc,
6136 <m-t> load      = default ]
6137 <bch> load      = bch-default ]
6138 <cmr> load      = cmr-OT1 ]
6139 <pad> load      = pad-default ]
6140 <pmn> load      = pmnj-default ]
6141 <ppl> load      = ppl-default ]
6142 <ptm> load      = ptm-default ]
6143 <m-t|bch|pad|pmn> { encoding = OT1,
6144 <cmr|ppl|ptm> { encoding = {OT1,OT4},
6145 <bch> family     = bch,
6146 <cmr> family     = cmr,
6147 <pad> family     = {pad,padx,padj},
6148 <pmn> family     = pmnj,
6149 <ppl> family     = {ppl,pplx,pplj},
6150 <ptm> family     = {ptm,ptmx,ptmj},
6151 shape          = sc }
6152 {
6153 a = {50,50},
6154 <cmr|pad|ppl|ptm> \ae = {50,  },
6155 <bch|pmn> c = {50,  },
6156 <bch|pad|pmn> d = { ,50},
6157 <m-t|bch|cmr|pad|pmn|ptm> f = { ,50},
6158 <bch|pad|pmn> g = {50,  },
6159 <m-t|cmr|pad|pmn|ppl|ptm> j = {50,  },
6160 <bch> j = {100,  },
6161 <m-t|bch|cmr|pad|pmn|ppl> l = { ,50},
6162 <ptm> l = { ,80},
6163 <m-t|bch|cmr|pad|pmn|ppl> 013 = { ,50}, % fl
6164 <ptm> 013 = { ,80}, % fl
6165 <bch|pad|pmn> o = {50,50},
6166 <pad|pmn> \oe = {50,  },
6167 <ppl> p = { 0, 0},
6168 <bch|pad|pmn> q = {50,70},

```

```

6169 <ppi>      q = { 0,  },
6170 <m-t|cmr|pad|pmn|ppi|ptm>      r = {  , 0},
6171      t = {50,50},
6172 <m-t|bch|cmr|pad|pmn|ppi>      y = {50,50}
6173 <ptm>      y = {80,80}
6174 }
6175
6176 \SetProtrusion
6177 <m-t> [ name      = T1-sc,
6178 <bch> [ name      = bch-sc-T1,
6179 <cmr> [ name      = cmr-sc-T1,
6180 <pad> [ name      = pad-sc-T1,
6181 <pmn> [ name      = pmnj-sc-T1,
6182 <ppi> [ name      = ppl-sc-T1,
6183 <ptm> [ name      = ptm-sc-T1,
6184 <m-t> load      = T1-default ]
6185 <bch> load      = bch-T1      ]
6186 <cmr> load      = cmr-T1      ]
6187 <pad> load      = pad-T1      ]
6188 <pmn> load      = pmnj-T1     ]
6189 <ppi> load      = ppl-T1      ]
6190 <ptm> load      = ptm-T1      ]
6191 { encoding = {T1,LY1},
6192 <bch> family = bch,
6193 <cmr> family = cmr,
6194 <pad> family = {pad,padx,padj},
6195 <pmn> family = pmnj,
6196 <ppi> family = {ppl,pplx,pplj},
6197 <ptm> family = {ptm,ptmx,ptmj},
6198 shape      = sc }
6199 {
6200   a = {50,50},
6201 <cmr|pad|ppi|ptm> \ae = {50,  },
6202 <bch|pmn> c = {50,  },
6203 <bch|pad|pmn> d = {  ,50},
6204 <m-t|bch|cmr|pad|pmn|ptm> f = {  ,50},
6205 <bch|pad|pmn> g = {50,  },
6206 <m-t|cmr|pad|pmn|ppi|ptm> j = {50,  },
6207 <bch> j = {100,  },
6208 <m-t|bch|cmr|pad|pmn|ppi> l = {  ,50},
6209 <ptm> l = {  ,80},
6210 <m-t|bch|cmr|pad|pmn|ppi> 029 = {  ,50}, % fl
6211 <ptm> 029 = {  ,80}, % fl
6212 <bch|pad|pmn> o = {50,50},
6213 <bch|pad|pmn> \oe = {50,  },
6214 <ppi> p = { 0, 0},
6215 <bch|pad|pmn> q = {50,70},
6216 <ppi> q = { 0,  },
6217 <m-t|cmr|pad|pmn|ppi|ptm> r = {  , 0},
6218      t = {50,50},
6219 <m-t|bch|cmr|pad|pmn|ppi> y = {50,50}
6220 <ptm> y = {80,80}
6221 }
6222
6223 <!(big|ugm)>
6224 <*m-t|cmr>
6225 \SetProtrusion
6226 <m-t> [ name      = T2A-sc,
6227 <cmr> [ name      = cmr-sc-T2A,
6228 <m-t> load      = T2A-default ]
6229 <cmr> load      = cmr-T2A      ]
6230 { encoding = T2A,
6231 <cmr> family = cmr,
6232 shape      = sc }
6233 {

```

```

6234     \cyra = {50,50},
6235     \cyrq = { ,50},
6236     \cyrt = {50,50},
6237     \cyrq = { ,50}
6238 }
6239
6240 </m-t|cmr>
6241 <*m-t>
6242 \SetProtrusion
6243 [ name      = QX-sc,
6244   load      = QX-default ]
6245 { encoding = QX,
6246   shape     = sc }
6247 {
6248   a = {50,50},
6249   f = { ,50},
6250   j = {50, },
6251   l = { ,50},
6252   013 = { ,50}, % fl
6253   r = { , 0},
6254   t = {50,50},
6255   y = {50,50}
6256 }
6257
6258 </m-t>
6259 <*cmr|bch>
6260 \SetProtrusion
6261 <bch> [ name      = bch-sc-T5,
6262 <bch>   load      = bch-T5 ]
6263 <cmr> [ name      = cmr-sc-T5,
6264 <cmr>   load      = cmr-T5 ]
6265 { encoding = T5,
6266 <bch>   family   = bch,
6267 <cmr>   family   = cmr,
6268   shape   = sc }
6269 {
6270   a = {50,50},
6271 <bch>   c = {50, },
6272 <bch>   d = { ,50},
6273   f = { ,50},
6274 <bch>   g = {50, },
6275 <bch>   j = {100, },
6276 <cmr>   j = {50, },
6277   l = { ,50},
6278 <bch>   o = {50,50},
6279 <bch>   q = { 0, },
6280 <cmr>   r = { , 0},
6281   t = {50,50},
6282   y = {50,50}
6283 }
6284
6285 </cmr|bch>
6286 <*pmn>
6287 \SetProtrusion
6288 [ name      = pmnx-sc,
6289   load      = pmnj-sc ]
6290 { encoding = OT1,
6291   family   = pmnx,
6292   shape     = sc }
6293 {
6294   l = {230,180}
6295 }
6296
6297 \SetProtrusion
6298 [ name      = pmnx-sc-T1,

```

```

6299     load      = pmnj-sc-T1 ]
6300     { encoding = {T1,LY1},
6301       family   = pmnx,
6302       shape    = sc }
6303     {
6304       1 = {230,180}
6305     }
6306

```

15.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's `fontinstallationguide` suggests `si`.

```

6307 \SetProtrusion
6308 [ name      = pmnj-scit,
6309   load      = pmnj-it   ]
6310 { encoding = OT1,
6311   family   = pmnj,
6312   shape    = {scit,si} }
6313 {
6314   a = {50, },
6315   \ae = { , -50},
6316   b = {20, -50},
6317   c = {50, -50},
6318   d = {20, 0},
6319   e = {20, -50},
6320   f = {10, 0},
6321   012 = {10, -50}, % fi
6322   013 = {10, -50}, % fl
6323   014 = {10, -50}, % ffi
6324   015 = {10, -50}, % ffl
6325   g = {50, -50},
6326   i = {20, -50},
6327   j = {20, 0},
6328   k = {20, },
6329   l = {20, 50},
6330   m = { , -30},
6331   n = { , -30},
6332   o = {50, },
6333   \oe = {50, -50},
6334   p = {20, -50},
6335   q = {50, },
6336   r = {20, 0},
6337   s = {20, -30},
6338   t = {70, },
6339   u = {50, -50},
6340   v = {100, },
6341   w = {100, },
6342   y = {50, },
6343   z = { , -50}
6344 }
6345
6346 \SetProtrusion
6347 [ name      = pmnj-scit-T1,
6348   load      = pmnj-it-T1 ]
6349 { encoding = {T1,LY1},
6350   family   = pmnj,
6351   shape    = {scit,si}   }
6352 {
6353   a = {50, },
6354   \ae = { , -50},
6355   b = {20, -50},
6356   c = {50, -50},
6357   d = {20, 0},

```

```

6358     e = {20,-50},
6359     f = {10, 0},
6360     028 = {10,-50}, % fi
6361     029 = {10,-50}, % fl
6362     030 = {10,-50}, % ffi
6363     031 = {10,-50}, % ffl
6364     g = {50,-50},
6365     i = {20,-50},
6366     188 = {20, 0}, % ij
6367     j = {20, 0},
6368     k = {20, },
6369     l = {20,50},
6370     m = { , -30},
6371     n = { , -30},
6372     o = {50, },
6373     \oe = {50,-50},
6374     p = {20,-50},
6375     q = {50, },
6376     r = {20, 0},
6377     s = {20,-30},
6378     t = {70, },
6379     u = {50,-50},
6380     v = {100, },
6381     w = {100, },
6382     y = {50, },
6383     z = { , -50}
6384 }
6385
6386 \SetProtrusion
6387 [ name      = pmnx-scit,
6388   load      = pmnj-scit ]
6389 { encoding = OT1,
6390   family   = pmnx,
6391   shape    = {scit,si} }
6392 {
6393   1 = {100,150}
6394 }
6395
6396 \SetProtrusion
6397 [ name      = pmnx-scit-T1,
6398   load      = pmnj-scit-T1 ]
6399 { encoding = {T1,LY1},
6400   family   = pmnx,
6401   shape    = {scit,si} }
6402 {
6403   1 = {100,150}
6404 }
6405
6406 </pmn>

```

15.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino. Anybody?

```

6407 \SetProtrusion
6408 <m-t> [ name      = textcomp ]
6409 <bch> [ name      = bch-textcomp ]
6410 <blg> [ name      = blg-textcomp ]
6411 <cmr> [ name      = cmr-textcomp ]
6412 <pad> [ name      = pad-textcomp ]
6413 <pmn> [ name      = pmn-textcomp ]
6414 <ppl> [ name      = ppl-textcomp ]
6415 <ptm> [ name      = ptm-textcomp ]
6416 <ugm> [ name      = ugm-textcomp ]

```

```

6417 <m-t> { encoding = TS1      }
6418 <!m-t> { encoding = TS1,
6419 <bch>    family   = bch }
6420 <blg>    family   = blg }
6421 <cmr>    family   = cmr }
6422 <pad>    family   = {pad,padx,padj} }
6423 <pmn>    family   = {pmnx,pmnj} }
6424 <ppl>    family   = {ppl,pplx,pplj} }
6425 <ptm>    family   = {ptm,ptmx,ptmj} }
6426 <ugm>    family   = ugm }
6427 {
6428 <blg>      \textquotestraightbase = {400,500},
6429 <cmr>      \textquotestraightbase = {300,300},
6430 <pad|pmn>  \textquotestraightbase = {400,400},
6431 <blg>      \textquotestraightdblbase = {300,400},
6432 <cmr|pmn>  \textquotestraightdblbase = {300,300},
6433 <pad>      \textquotestraightdblbase = {400,400},
6434 <bch|cmr|pad|pmn|ugm> \texttwelveudash = {200,200},
6435 <bch|cmr|pad|pmn> \textthreequartersemdash = {150,150},
6436 <ugm>      \textthreequartersemdash = {200,200},
6437 <blg>      \textquotesingle = {500,600},
6438 <cmr|pmn>  \textquotesingle = {300,400},
6439 <pad>      \textquotesingle = {400,500},
6440 <ptm>      \textquotesingle = {500,500},
6441 <ugm>      \textquotesingle = {300,500},
6442 <bch|cmr|pmn> \textasteriskcentered = {200,300},
6443 <blg>      \textasteriskcentered = {150,200},
6444 <pad>      \textasteriskcentered = {300,300},
6445 <ugm>      \textasteriskcentered = {100,200},
6446 <pmn>      \textfractionsolidus = {-200,-200},
6447 <cmr>      \textoneoldstyle = {100,100},
6448 <pmn>      \textoneoldstyle = { , 50},
6449 <cmr>      \textthreeoldstyle = { , 50},
6450 <pad|pmn>  \textthreeoldstyle = { 50, },
6451 <cmr>      \textfouroldstyle = { 50, 50},
6452 <pad|pmn>  \textfouroldstyle = { 50, },
6453 <cmr|pad|pmn> \textsevenoldstyle = { 50, 80},
6454 <cmr>      \textlangle = {400, },
6455 <cmr>      \textrangle = { , 400},
6456 <m-t|bch|pmn|ptm> \textminus = {200,200},
6457 <cmr|pad|ppl> \textminus = {300,300},
6458 <blg|ugm> \textminus = {250,300},
6459 <bch|pad|pmn> \textlbrackdbl = {100, },
6460 <blg>      \textlbrackdbl = {200, },
6461 <bch|pad|pmn> \textrbrackdbl = { , 100},
6462 <blg>      \textrbrackdbl = { , 200},
6463 <pmn>      \textasciigrave = {200,500},
6464 <bch|blg|cmr|pad|pmn> \texttildelow = {200,250},
6465 <pmn>      \textasciibreve = {300,400},
6466 <pmn>      \textasciicaron = {300,400},
6467 <pmn>      \textacutedbl = {200,300},
6468 <pmn>      \textgravedbl = {150,300},
6469 <bch|pmn|ugm> \textdagger = { 80, 80},
6470 <blg>      \textdagger = {200,200},
6471 <cmr|pad> \textdagger = {100,100},
6472 <ptm>      \textdagger = {150,150},
6473 <blg>      \textdaggerdbl = {150,150},
6474 <cmr|pad|pmn> \textdaggerdbl = { 80, 80},
6475 <ptm>      \textdaggerdbl = {100,100},
6476 <bch>      \textbardbl = {100,100},
6477 <blg|ugm> \textbardbl = {150,150},
6478 <bch>      \textbullet = {200,200},
6479 <blg>      \textbullet = {400,500},
6480 <cmr|pad|pmn> \textbullet = { , 100},
6481 <ptm>      \textbullet = {150,150},

```



```

6482 <ugm> \textbullet = { 50,100},
6483 <bch|cmr|pmn> \textcelsius = { 50, },
6484 <pad> \textcelsius = { 80, },
6485 <bch> \textflorin = { 50, 50},
6486 <blg> \textflorin = {100,100},
6487 <pad|ugm> \textflorin = { ,100},
6488 <pmn> \textflorin = { 50,100},
6489 <ptm> \textflorin = { 50, 70},
6490 <cmr> \textcolonmonetary = { , 50},
6491 <pad|pmn> \textcolonmonetary = { 50, },
6492 <pmn> \textinterrobang = { ,100},
6493 <pmn> \textinterrobangdown = {100, },
6494 <m-t|pad|ptm> \texttrademark = {100,100},
6495 <bch> \texttrademark = {150,150},
6496 <blg|cmr|ppl> \texttrademark = {200,200},
6497 <pmn> \texttrademark = { 50, 50},
6498 <ugm> \texttrademark = {100,150},
6499 <bch|ugm> \textcent = { 50, },
6500 <ptm> \textcent = {100,100},
6501 <bch> \textsterling = { 50, },
6502 <ugm> \textsterling = { , 50},
6503 <bch> \textbrokenbar = {200,200},
6504 <blg> \textbrokenbar = {250,250},
6505 <ugm> \textbrokenbar = {200,300},
6506 <pmn> \textasciidieresis = {300,400},
6507 <m-t|bch|cmr|pad|ptm|ugm> \textcopyright = {100,100},
6508 <pmn> \textcopyright = {100,150},
6509 <ppl> \textcopyright = {200,200},
6510 <bch|cmr|ugm> \textordfeminine = {100,200},
6511 <pad|pmn> \textordfeminine = {200,200},
6512 <bch|cmr|pad|pmn|ugm> \textlnot = {200, },
6513 <blg> \textlnot = {200,100},
6514 <m-t|bch|cmr|pad|ptm|ugm> \textregistered = {100,100},
6515 <pmn> \textregistered = { 50,150},
6516 <ppl> \textregistered = {200,200},
6517 <pmn> \textasciimacron = {150,200},
6518 <m-t|ppl|ptm> \textdegree = {300,300},
6519 <bch> \textdegree = {150,200},
6520 <blg|ugm> \textdegree = {200,200},
6521 <cmr|pad> \textdegree = {400,400},
6522 <pmn> \textdegree = {150,400},
6523 <bch|cmr|pad|pmn|ugm> \textpm = {150,200},
6524 <blg> \textpm = {100,100},
6525 <ptm> \textpm = { 50, 80},
6526 <bch|blg|ugm> \texttwosuperior = {100,200},
6527 <cmr> \texttwosuperior = { 50,100},
6528 <pad|pmn> \texttwosuperior = {200,200},
6529 <ptm> \texttwosuperior = { 50, 50},
6530 <bch|blg|ugm> \textthreesuperior = {100,200},
6531 <cmr> \textthreesuperior = { 50,100},
6532 <pad|pmn> \textthreesuperior = {200,200},
6533 <ptm> \textthreesuperior = { 50, 50},
6534 <pmn> \textasciicute = {300,400},
6535 <bch|ugm> \textmu = { ,100},
6536 <bch|pad|pmn> \textparagraph = { ,100},
6537 <bch|cmr|pad|pmn> \textperiodcentered = {300,400},
6538 <blg> \textperiodcentered = {400,500},
6539 <ptm> \textperiodcentered = {300,300},
6540 <ugm> \textperiodcentered = {200,500},
6541 <bch|blg|ugm> \textonesuperior = {200,300},
6542 <cmr|pad|pmn> \textonesuperior = {200,200},
6543 <ptm> \textonesuperior = {100,100},
6544 <bch|pad|pmn|ugm> \textordmasculine = {200,200},
6545 <blg|cmr> \textordmasculine = {100,200},
6546 <bch|cmr|pmn> \texteuro = {100, },

```

```

6547 <pad> \texteuro = { 50,100},
6548 <bch> \texttimes = {200,200},
6549 <blg|ptm> \texttimes = {100,100},
6550 <cmr> \texttimes = {150,250},
6551 <pad> \texttimes = {100,150},
6552 <pmn> \texttimes = { 70,100},
6553 <ugm> \texttimes = {200,300},
6554 <bch|pad|pmn> \textdiv = {150,200}
6555 <blg> \textdiv = {100,100}
6556 <cmr> \textdiv = {150,250}
6557 <ptm> \textdiv = { 50,100},
6558 <ugm> \textdiv = {200,300},
6559 <ptm> \textperthousand = { ,50}
6560 <ugm> \textsection = { ,100},
6561 <ugm> \textonehalf = { 50,100},
6562 <ugm> \textonequarter = { 50,100},
6563 <ugm> \textthreequarters = { 50,100},
6564 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

6565 }
6566
6567 <*cmr|pad|pmn|ugm>
6568 \SetProtrusion
6569 <cmr> [ name = cmr-textcomp-it ]
6570 <pad> [ name = pad-textcomp-it ]
6571 <pmn> [ name = pmn-textcomp-it ]
6572 <ugm> [ name = ugm-textcomp-it ]
6573 { encoding = TS1,
6574 <cmr> family = cmr,
6575 <pad> family = {pad,padx,padj},
6576 <pmn> family = {pmnx,pmnj},
6577 <ugm> family = ugm,
6578 <!ugm> shape = {it,s1} }
6579 <ugm> shape = it }
6580 {
6581 <cmr> \textquotestraightbase = {300,600},
6582 <pad|pmn> \textquotestraightbase = {400,400},
6583 <cmr> \textquotestraightdblbase = {300,600},
6584 <pad> \textquotestraightdblbase = {300,400},
6585 <pmn> \textquotestraightdblbase = {300,300},
6586 \texttwelveudash = {200,200},
6587 <cmr|pad|pmn> \textthreequartersemdash = {150,150},
6588 <ugm> \textthreequartersemdash = {200,200},
6589 <cmr> \textquotesingle = {600,300},
6590 <pad> \textquotesingle = {800,100},
6591 <pmn> \textquotesingle = {300,200},
6592 <ugm> \textquotesingle = {500,500},
6593 <cmr> \textasteriskcentered = {300,200},
6594 <pad> \textasteriskcentered = {500,100},
6595 <pmn> \textasteriskcentered = {200,300},
6596 <ugm> \textasteriskcentered = {300,150},
6597 <pmn> \textfrown = {-200,-200},
6598 <cmr> \textoneoldstyle = {100, 50},
6599 <pad> \textoneoldstyle = {100, },
6600 <pmn> \textoneoldstyle = { 50, },
6601 <pad> \texttwooldstyle = { 50, },
6602 <pmn> \texttwooldstyle = {-50, },
6603 <cmr> \textthreeoldstyle = {100, 50},
6604 <pmn> \textthreeoldstyle = {-100, },
6605 <cmr> \textfouroldstyle = { 50, 50},
6606 <pad> \textfouroldstyle = { 50,100},
6607 <cmr> \textsevenoldstyle = { 50, 80},
6608 <pad> \textsevenoldstyle = { 50, },
6609 <pmn> \textsevenoldstyle = { 20, },

```

```

6610 <cmr> \textlangle = {400, },
6611 <cmr> \textrangle = { ,400},
6612 <cmr|pad> \textminus = {300,300},
6613 <pmn> \textminus = {200,200},
6614 <ugm> \textminus = {250,300},
6615 <pad|pmn> \textlbrackdbl = {100, },
6616 <pad|pmn> \textrbrackdbl = { ,100},
6617 <pmn> \textasciigrave = {300,300},
6618 <cmr|pad|pmn> \texttildelow = {200,250},
6619 <pmn> \textasciibreve = {300,300},
6620 <pmn> \textasciicaron = {300,300},
6621 <pmn> \textacutedbl = {200,300},
6622 <pmn> \textgravedbl = {150,300},
6623 <cmr> \textdagger = {100,100},
6624 <pad> \textdagger = {200,100},
6625 <pmn> \textdagger = { 80, 50},
6626 <ugm> \textdagger = { 80, 80},
6627 <cmr|pad> \textdaggerdbl = { 80, 80},
6628 <pmn> \textdaggerdbl = { 80, 50},
6629 <ugm> \textbardbl = {150,150},
6630 <cmr> \textbullet = {200,100},
6631 <pad> \textbullet = {300, },
6632 <pmn> \textbullet = { 30, 70},
6633 <ugm> \textbullet = { 50,100},
6634 <cmr> \textcelsius = {100, },
6635 <pad> \textcelsius = {200, },
6636 <pmn> \textcelsius = { 50,-50},
6637 <pad> \textflorin = {100, },
6638 <pmn> \textflorin = { 50,100},
6639 <ugm> \textflorin = { ,100},
6640 <cmr> \textcolonmonetary = {150, },
6641 <pad> \textcolonmonetary = {100, },
6642 <pmn> \textcolonmonetary = { 50,-50},
6643 <cmr|pad> \texttrademark = {200, },
6644 <pmn> \texttrademark = { 50,100},
6645 <ugm> \texttrademark = {150, 50},
6646 <ugm> \textcent = { 50, },
6647 <ugm> \textsterling = { , 50},
6648 <ugm> \textbrokenbar = {200,300},
6649 <pmn> \textasciidieresis = {300,200},
6650 <cmr> \textcopyright = {100, },
6651 <pad> \textcopyright = {200,100},
6652 <pmn> \textcopyright = {100,150},
6653 <ugm> \textcopyright = {300, },
6654 <cmr> \textordfeminine = {100,100},
6655 <pmn> \textordfeminine = {200,200},
6656 <ugm> \textordfeminine = {100,200},
6657 <cmr|pad> \textlnot = {300, },
6658 <pmn|ugm> \textlnot = {200, },
6659 <cmr> \textregistered = {100, },
6660 <pad> \textregistered = {200,100},
6661 <pmn> \textregistered = { 50,150},
6662 <ugm> \textregistered = {300, },
6663 <pmn> \textasciimacron = {150,200},
6664 <cmr|pad> \textdegree = {500,100},
6665 <pmn> \textdegree = {150,150},
6666 <ugm> \textdegree = {300,200},
6667 <cmr> \textpm = {150,100},
6668 <pad> \textpm = {200,150},
6669 <pmn|ugm> \textpm = {150,200},
6670 <cmr> \textonesuperior = {400, },
6671 <pad> \textonesuperior = {300,100},
6672 <pmn> \textonesuperior = {200,100},
6673 <ugm> \textonesuperior = {300,300},
6674 <cmr> \texttwosuperior = {400, },

```

```

6675 <pad> \texttwosuperior = {300, },
6676 <pmn> \texttwosuperior = {200,100},
6677 <ugm> \texttwosuperior = {300,200},
6678 <cmr> \textthreesuperior = {400, },
6679 <pad> \textthreesuperior = {300, },
6680 <pmn> \textthreesuperior = {200,100},
6681 <ugm> \textthreesuperior = {300,200},
6682 <ugm> \textmu = { ,100},
6683 <pmn> \textasciicute = {300,200},
6684 <cmr> \textparagraph = {200, },
6685 <pmn> \textparagraph = { ,100},
6686 <cmr> \textperiodcentered = {500,500},
6687 <pad|pmn|ugm> \textperiodcentered = {300,400},
6688 <cmr> \textordmasculine = {100,100},
6689 <pmn> \textordmasculine = {200,200},
6690 <ugm> \textordmasculine = {300,200},
6691 <cmr> \texteuro = {200, },
6692 <pad> \texteuro = {100, },
6693 <pmn> \texteuro = {100,-50},
6694 <cmr> \texttimes = {200,200},
6695 <pad> \texttimes = {200,100},
6696 <pmn> \texttimes = { 70,100},
6697 <ugm> \texttimes = {200,300},
6698 <cmr|pad> \textdiv = {200,200},
6699 <pmn> \textdiv = {150,200},
6700 <ugm> \textdiv = {200,300},
6701 <ugm> \textsection = { ,200},
6702 <ugm> \textonehalf = { 50,100},
6703 <ugm> \textonequarter = { 50,100},
6704 <ugm> \textthreequarters = { 50,100},
6705 <ugm> \textsurd = { ,100}
6706 }
6707
6708 </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}

```

```

6709 <*cmr>
6710 \SetProtrusion
6711 [ name = cmr-math-letters ]
6712 { encoding = OML,
6713   family = cmm,
6714   series = {m,b},
6715   shape = it }
6716 {
6717   A = {100, 50}, % \mathnormal
6718   B = { 50, },

```

```

6719     C = { 50,  },
6720     D = { 50, 50},
6721     E = { 50,  },
6722     F = {100, 50},
6723     G = { 50, 50},
6724     H = { 50, 50},
6725     I = { 50, 50},
6726     J = {150, 50},
6727     K = { 50,100},
6728     L = { 50, 50},
6729     M = { 50,  },
6730     N = { 50,  },
6731     O = { 50,  },
6732     P = { 50,  },
6733     Q = { 50, 50},
6734     R = { 50,  },
6735     S = { 50,  },
6736     T = { 50,100},
6737     U = { 50, 50},
6738     V = {100,100},
6739     W = { 50,100},
6740     X = { 50,100},
6741     Y = {100,100},
6742     f = {100,100},
6743     h = {  ,100},
6744     i = {  , 50},
6745     j = {  , 50},
6746     k = {  , 50},
6747     r = {  , 50},
6748     v = {  , 50},
6749     w = {  , 50},
6750     x = {  , 50},
6751     "0B = { 50,100}, % \alpha
6752     "0C = { 50, 50}, % \beta
6753     "0D = {200,150}, % \gamma
6754     "0E = { 50, 50}, % \delta
6755     "0F = { 50, 50}, % \epsilon
6756     "10 = { 50,150}, % \zeta
6757     "12 = { 50,  }, % \theta
6758     "13 = {  ,100}, % \iota
6759     "14 = {  ,100}, % \kappa
6760     "15 = {100, 50}, % \lambda
6761     "16 = {  , 50}, % \mu
6762     "17 = {  , 50}, % \nu
6763     "18 = {  , 50}, % \xi
6764     "19 = { 50,100}, % \pi
6765     "1A = { 50, 50}, % \rho
6766     "1B = {  ,150}, % \sigma
6767     "1C = { 50,150}, % \tau
6768     "1D = { 50, 50}, % \upsilon
6769     "1F = { 50,100}, % \chi
6770     "20 = { 50, 50}, % \psi
6771     "21 = {  , 50}, % \omega
6772     "22 = {  , 50}, % \varepsilon
6773     "23 = {  , 50}, % \vartheta
6774     "24 = {  , 50}, % \varpi
6775     "25 = {100,  }, % \varrho
6776     "26 = {100,100}, % \varsigma
6777     "27 = { 50, 50}, % \varphi
6778     "28 = {100,100}, % \leftharpoonup
6779     "29 = {100,100}, % \leftharpoondown
6780     "2A = {100,100}, % \rightharpoonup
6781     "2B = {100,100}, % \rightharpoondown
6782     "2C = {300,200}, % \hook
6783     "2D = {200,300}, % \rhook

```

```

6784 "2E = { ,100}, % \triangleright
6785 "2F = {100, }, % \triangleleft
6786 "3A = { ,500}, % ., \ldotp
6787 "3B = { ,500}, % ,
6788 "3C = {200,100}, % <
6789 "3D = {300,400}, % /
6790 "3E = {100,200}, % >
6791 "3F = {200,200}, % \star
6792 "5B = { ,100}, % \flat
6793 "5E = {200,200}, % \smile
6794 "5F = {200,200}, % \frown
6795 "7C = {100, }, % \jmath
6796 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

6797 }
6798

```

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}

```

```

6799 \SetProtrusion
6800 [ name = cmr-math-symbols ]
6801 { encoding = OMS,
6802   family = cmsy,
6803   series = {m,b},
6804   shape = n }
6805 {
6806   A = {150, 50}, % \mathcal
6807   C = { ,100},
6808   D = { , 50},
6809   F = { 50,150},
6810   I = { ,100},
6811   J = {100,150},
6812   K = { ,100},
6813   L = {100, },
6814   M = { 50, 50},
6815   N = { 50,100},
6816   P = { , 50},
6817   Q = { 50, },
6818   R = { , 50},
6819   T = { 50,150},
6820   V = { 50, 50},
6821   W = { , 50},
6822   X = {100,100},
6823   Y = {100, },
6824   Z = {100,150},
6825   "00 = {300,300}, % -
6826   "01 = { ,700}, % \cdot, \cdotp
6827   "02 = {150,250}, % \times
6828   "03 = {150,250}, % *, \ast
6829   "04 = {200,300}, % \div
6830   "05 = {150,250}, % \diamond
6831   "06 = {200,200}, % \pm
6832   "07 = {200,200}, % \mp
6833   "08 = {100,100}, % \oplus
6834   "09 = {100,100}, % \ominus
6835   "0A = {100,100}, % \otimes
6836   "0B = {100,100}, % \oslash
6837   "0C = {100,100}, % \odot
6838   "0D = {100,100}, % \bigcirc
6839   "0E = {100,100}, % \circ
6840   "0F = {100,100}, % \bullet
6841   "10 = {100,100}, % \asymp

```

```

6842 "11 = {100,100}, % \equiv
6843 "12 = {200,100}, % \subseteq
6844 "13 = {100,200}, % \supseteq
6845 "14 = {200,100}, % \leq
6846 "15 = {100,200}, % \geq
6847 "16 = {200,100}, % \preceq
6848 "17 = {100,200}, % \succeq
6849 "18 = {200,200}, % \sim
6850 "19 = {150,150}, % \approx
6851 "1A = {200,100}, % \subset
6852 "1B = {100,200}, % \supset
6853 "1C = {200,100}, % \ll
6854 "1D = {100,200}, % \gg
6855 "1E = {300,100}, % \prec
6856 "1F = {100,300}, % \succ
6857 "20 = {100,200}, % \leftarrow
6858 "21 = {200,100}, % \rightarrow
6859 "22 = {100,100}, % \uparrow
6860 "23 = {100,100}, % \downarrow
6861 "24 = {100,100}, % \leftrightarrows
6862 "25 = {100,100}, % \nearrow
6863 "26 = {100,100}, % \searrow
6864 "27 = {100,100}, % \simeq
6865 "28 = {100,100}, % \Leftarrow
6866 "29 = {100,100}, % \Rightarrow
6867 "2A = {100,100}, % \Uparrow
6868 "2B = {100,100}, % \Downarrow
6869 "2C = {100,100}, % \Leftrightarrow
6870 "2D = {100,100}, % \nrightarrow
6871 "2E = {100,100}, % \swarrow
6872 "2F = { ,100}, % \propto
6873 "30 = { ,400}, % \prime
6874 "31 = {100,100}, % \infty
6875 "32 = {150,100}, % \ln
6876 "33 = {100,150}, % \ni
6877 "34 = {100,100}, % \triangle, \bigtriangleup
6878 "35 = {100,100}, % \bigtriangledown
6879 "38 = { ,100}, % \forall
6880 "39 = {100, }, % \exists
6881 "3A = {200, }, % \neg
6882 "3E = {200,200}, % \top
6883 "3F = {200,200}, % \bot, \perp
6884 "5E = {100,200}, % \wedge
6885 "5F = {100,200}, % \vee
6886 "60 = { ,300}, % \vdash
6887 "61 = {300, }, % \dashv
6888 "62 = {100,100}, % \lfloor
6889 "63 = {100,100}, % \rfloor
6890 "64 = {100,100}, % \lceil
6891 "65 = {100,100}, % \rceil
6892 "66 = {150, }, % \lbrace
6893 "67 = { ,150}, % \rbrace
6894 "68 = {400, }, % \langle
6895 "69 = { ,400}, % \rangle
6896 "6C = {100,100}, % \updownarrow
6897 "6D = {100,100}, % \Updownarrow
6898 "6E = {100,300}, % \, \backslash, \setminus
6899 "72 = {100,100}, % \nabla
6900 "79 = {200,200}, % \dagger
6901 "7A = {100,100}, % \ddagger
6902 "7B = {100, }, % \mathparagraph
6903 "7C = {100,100}, % \clubsuit
6904 "7D = {100,100}, % \diamondsuit
6905 "7E = {100,100}, % \heartsuit
6906 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```
6907 }
6908
```

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}
```

```
6909 </cmr>
6910 </cfg-t>
```

15.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
6911 <*cfg-u>
```

Symbol font 'a'.

```
6912 <*msa>
6913 \SetProtrusion
6914 [ name = AMS-a ]
6915 { encoding = U,
6916   family = msa }
6917 {
6918   "05 = {150,250}, % \centerdot
6919   "06 = {100,100}, % \lozenge
6920   "07 = { 50, 50}, % \blacklozenge
6921   "08 = { 50, 50}, % \circlearrowright
6922   "09 = { 50, 50}, % \circlearrowleft
6923   "0A = {100,100}, % \rightleftharpoons
6924   "0B = {100,100}, % \leftrightharpoons
6925   "0D = {-50,200}, % \Vdash
6926   "0E = {-50,200}, % \Vvdash
6927   "0F = {-70,150}, % \vDash
6928   "10 = {100,150}, % \twoheadrightarrow
6929   "11 = {100,150}, % \twoheadleftarrow
6930   "12 = { 50,100}, % \leftleftarrows
6931   "13 = { 50, 80}, % \rightrightarrows
6932   "14 = {120,120}, % \upuparrows
6933   "15 = {120,120}, % \downdownarrows
6934   "16 = {200,200}, % \upharpoonright
6935   "17 = {200,200}, % \downharpoonright
6936   "18 = {200,200}, % \upharpoonleft
6937   "19 = {200,200}, % \downharpoonleft
6938   "1A = { 80,100}, % \rightarrowtail
6939   "1B = { 80,100}, % \leftarrowtail
6940   "1C = { 50, 50}, % \leftrightarrows
6941   "1D = { 50, 50}, % \rightleftarrows
6942   "1E = {250, }, % \Lsh
6943   "1F = { ,250}, % \Rsh
6944   "20 = {100,100}, % \rightsquigarrow
6945   "21 = {100,100}, % \leftrightsquigarrow
6946   "22 = {100, 50}, % \looparrowleft
6947   "23 = { 50,100}, % \looparrowright
6948   "24 = { 50, 80}, % \circeq
6949   "25 = { ,100}, % \succsim
6950   "26 = { ,100}, % \gtrsim
6951   "27 = { ,100}, % \gtrapprox
6952   "28 = {150, 50}, % \multimap
6953   "2B = {100,150}, % \doteqdot
6954   "2C = {100,150}, % \triangleq
6955   "2D = {100, 50}, % \precsim
6956   "2E = {100, 50}, % \lesssim
6957   "2F = { 50, 50}, % \lessapprox
```



```

6958 "30 = {100, 50}, % \eqslantless
6959 "31 = { 50, 50}, % \eqslantgtr
6960 "32 = {100, 50}, % \curlyeqprec
6961 "33 = { 50,100}, % \curlyeqsucc
6962 "34 = {100, 50}, % \preccurlyeq
6963 "36 = { 50,  }, % \leqslant
6964 "38 = {  , 50}, % \backprime
6965 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
6966 "3C = { 50,100}, % \succcurlyeq
6967 "3E = {  , 50}, % \geqslant
6968 "40 = {  , 50}, % \sqsubset
6969 "41 = { 50,  }, % \sqsupset
6970 "42 = {  ,150}, % \vartriangleright, \rhd
6971 "43 = {150,  }, % \vartriangleleft, \lhd
6972 "44 = {  ,100}, % \trianglerighteq, \unrhd
6973 "45 = {100,  }, % \trianglelefteq, \unlhd
6974 "46 = {100,100}, % \bigstar
6975 "48 = { 50, 50}, % \blacktriangledown
6976 "49 = {  ,100}, % \blacktriangleright
6977 "4A = {100,  }, % \blacktriangleleft
6978 "4B = {  ,150}, % \dashrightarrow (the arrow)
6979 "4C = {150,  }, % \dashleftarrow
6980 "4D = { 50, 50}, % \vartriangle
6981 "4E = { 50, 50}, % \blacktriangle
6982 "4F = { 50, 50}, % \triangledown
6983 "50 = { 50, 50}, % \eqcirc
6984 "56 = {  ,150}, % \Rrightarrow
6985 "57 = {150,  }, % \Lleftarrow
6986 "58 = {100,300}, % \checkmark
6987 "5C = { 50, 50}, % \angle
6988 "5D = { 50, 50}, % \measuredangle
6989 "5E = { 50, 50}, % \sphericalangle
6990 "5F = {  , 50}, % \varpropto
6991 "60 = {100,100}, % \smallsmile
6992 "61 = {100,100}, % \smallfrown
6993 "62 = { 50,  }, % \Subset
6994 "63 = {  , 50}, % \Supset
6995 "66 = {150,150}, % \curlywedge
6996 "67 = {150,150}, % \curlyvee
6997 "68 = { 50,150}, % \leftthreetimes
6998 "69 = {100, 50}, % \rightthreetimes
6999 "6C = { 50, 50}, % \bumpeq
7000 "6D = { 50, 50}, % \Bumpeq
7001 "6E = {100,  }, % \lll
7002 "6F = {  ,100}, % \ggg
7003 "70 = { 50,100}, % \ulcorner
7004 "71 = {100, 50}, % \urcorner
7005 "75 = {150,200}, % \dotplus
7006 "76 = { 50,100}, % \backsim
7007 "78 = { 50,100}, % \llcorner
7008 "79 = {100, 50}, % \lrcorner
7009 "7C = {100,100}, % \intercal
7010 "7D = { 50, 50}, % \circledcirc
7011 "7E = { 50, 50}, % \circledast
7012 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

7013 }
7014
7015 </msa>

```

Symbol font 'b'.

```

7016 <*msb>
7017 \SetProtrusion
7018 [ name = AMS-b ]

```

```

7019 { encoding = U,
7020     family   = msb }
7021 {
7022     A = { 50, 50}, % \mathbb
7023     C = { 50, 50},
7024     G = {   , 50},
7025     L = {   , 50},
7026     P = {   , 50},
7027     R = {   , 50},
7028     T = {   , 50},
7029     V = { 50, 50},
7030     X = { 50, 50},
7031     Y = { 50, 50},
7032     "00 = { 50, 50}, % \lvertneqq
7033     "01 = { 50, 50}, % \gvertneqq
7034     "02 = { 50, 50}, % \nleq
7035     "03 = { 50, 50}, % \ngeq
7036     "04 = {100, 50}, % \nless
7037     "05 = { 50,150}, % \ngtr
7038     "06 = {100, 50}, % \nprec
7039     "07 = { 50,150}, % \nsucc
7040     "08 = { 50, 50}, % \lneqq
7041     "09 = { 50, 50}, % \gneqq
7042     "0A = {100,100}, % \nleqslant
7043     "0B = {100,100}, % \ngeqslant
7044     "0C = {100, 50}, % \lneq
7045     "0D = { 50,100}, % \gneq
7046     "0E = {100, 50}, % \npreceq
7047     "0F = { 50,100}, % \nsucceq
7048     "10 = { 50,   }, % \precnsim
7049     "11 = { 50, 50}, % \succnsim
7050     "12 = { 50, 50}, % \lnsim
7051     "13 = { 50, 50}, % \gnsim
7052     "14 = { 50, 50}, % \lneqq
7053     "15 = { 50, 50}, % \ngeqq
7054     "16 = { 50, 50}, % \precneqq
7055     "17 = { 50, 50}, % \succneqq
7056     "18 = { 50, 50}, % \precnapprox
7057     "19 = { 50, 50}, % \succnapprox
7058     "1A = { 50, 50}, % \lnapprox
7059     "1B = { 50, 50}, % \gnapprox
7060     "1C = {150,200}, % \nsim
7061     "1D = { 50, 50}, % \ncong
7062     "1E = {100,150}, % \diagup
7063     "1F = {100,150}, % \diagdown
7064     "20 = {100, 50}, % \varsubsetneq
7065     "21 = { 50,100}, % \varsupsetneq
7066     "22 = {100, 50}, % \subsetneqq
7067     "23 = { 50,100}, % \supsetneqq
7068     "24 = {100, 50}, % \subsetneqq
7069     "25 = { 50,100}, % \supsetneqq
7070     "26 = {100, 50}, % \varsubsetneqq
7071     "27 = { 50,100}, % \varsupsetneqq
7072     "28 = {100, 50}, % \subsetneq
7073     "29 = { 50,100}, % \supsetneq
7074     "2A = {100, 50}, % \subseteq
7075     "2B = { 50,100}, % \supseteq
7076     "2C = { 50,100}, % \nparallel
7077     "2D = {100,150}, % \nmid
7078     "2E = {150,150}, % \nshortmid
7079     "2F = {100,100}, % \nshortparallel
7080     "30 = {   ,150}, % \nvdash
7081     "31 = {   ,150}, % \nVdash
7082     "32 = {   ,100}, % \nvDash
7083     "33 = {   ,100}, % \nVDash

```

```

7084 "34 = { ,100}, % \ntrianglerighteq
7085 "35 = {100, }, % \ntrianglelefteq
7086 "36 = {100, }, % \ntriangleleft
7087 "37 = { ,100}, % \ntriangleright
7088 "38 = {100,200}, % \nleftarrow
7089 "39 = {100,200}, % \nrightrightarrow
7090 "3A = {100,100}, % \nLeftarrow
7091 "3B = { 50,100}, % \nrightarrow
7092 "3C = {100,100}, % \nLefttrightharpoon
7093 "3D = {100,200}, % \nleftrightharpoon
7094 "3E = { 50, 50}, % \divideontimes
7095 "3F = { 50, 50}, % \varnothing
7096 "60 = {200, }, % \Finv
7097 "61 = { , 50}, % \Game
7098 "68 = {100,100}, % \eqsim
7099 "69 = { 50, }, % \beth
7100 "6A = { 50, }, % \gimel
7101 "6B = {150, }, % \daleth
7102 "6C = {200, }, % \lessdot
7103 "6D = { ,200}, % \gtrdot
7104 "6E = {100,200}, % \ltimes
7105 "6F = {150,100}, % \rtimes
7106 "70 = { 50,100}, % \shortmid
7107 "71 = { 50, 50}, % \shortparallel
7108 "72 = {200,300}, % \smallsetminus
7109 "73 = {100,200}, % \thicksim
7110 "74 = { 50,100}, % \thickapprox
7111 "75 = { 50, 50}, % \approx
7112 "76 = { 50,100}, % \succapprox
7113 "77 = { 50, 50}, % \precapprox
7114 "78 = {100,100}, % \curvearrowleft
7115 "79 = { 50,150}, % \curvearrowright
7116 "7A = { 50,200}, % \digamma
7117 "7B = {100, 50}, % \varkappa
7118 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

7119 }
7120
7121 (/msb)

```

15.8.8 Euler

Euler Roman font (package `euler`).

```

7122 (*eur)
7123 \SetProtrusion
7124 [ name = euler ]
7125 { encoding = U,
7126 family = eur }
7127 {
7128 "01 = {100,100},
7129 "03 = {100,150},
7130 "06 = { ,100},
7131 "07 = {100,150},
7132 "08 = {100,100},
7133 "0A = {100,100},
7134 "0B = { , 50},
7135 "0C = { ,100},
7136 "0D = {100,100},
7137 "0E = { ,100},
7138 "0F = {100,100},
7139 "10 = {100,100},
7140 "13 = { ,100},
7141 "14 = { ,100},

```

```

7142     "15 = {    , 50},
7143     "16 = {    , 50},
7144     "17 = { 50,100},
7145     "18 = { 50,100},
7146     "1A = {    , 50},
7147     "1B = {    , 50},
7148     "1C = { 50,100},
7149     "1D = { 50,100},
7150     "1E = { 50,100},
7151     "1F = { 50,100},
7152     "20 = {    , 50},
7153     "21 = {    , 50},
7154     "22 = { 50,100},
7155     "24 = {    , 50},
7156     "27 = { 50,100},
7157     1  = {100,100},
7158     7  = { 50,100},
7159     "3A = {300,500},
7160     "3B = {200,400},
7161     "3C = {200,100},
7162     "3D = {200,200},
7163     "3E = {100,200},
7164     A  = {    ,100},
7165     D  = {    , 50},
7166     J  = { 50,   },
7167     K  = {    , 50},
7168     L  = {    , 50},
7169     Q  = {    , 50},
7170     T  = { 50,   },
7171     X  = { 50, 50},
7172     Y  = { 50,   },
7173     h  = {    , 50},
7174     k  = {    , 50}
7175 }
7176

```

Extended by the `eulervm` package.

```

7177 \SetProtrusion
7178 [ name      = euler-vm,
7179   load      = euler ]
7180 { encoding = U,
7181   family   = zeur }
7182 {
7183     "28 = {100,200},
7184     "29 = {100,200},
7185     "2A = {100,150},
7186     "2B = {100,150},
7187     "2C = {200,300},
7188     "2D = {200,300},
7189     "2E = {    ,100},
7190     "2F = {100,   },
7191     "3F = {150,150},
7192     "5B = {    ,100},
7193     "5E = {100,100},
7194     "5F = {100,100},
7195     "80 = {    , 50},
7196     "81 = {200,250},
7197     "82 = {100,200}
7198 }
7199
7200 /eur

```

Euler Script font (`euca1`).

```

7201 *eus
7202 \SetProtrusion

```

```

7203 [ name      = euscript ]
7204 { encoding = U,
7205   family   = eus  }
7206 {
7207   A = {100,100},
7208   B = { 50,100},
7209   C = { 50, 50},
7210   D = { 50,100},
7211   E = { 50,100},
7212   F = { 50,  },
7213   G = { 50,  },
7214   H = {  ,100},
7215   K = {  , 50},
7216   L = {  ,150},
7217   M = {  , 50},
7218   N = {  , 50},
7219   O = { 50, 50},
7220   P = { 50, 50},
7221   T = {  ,100},
7222   U = {  , 50},
7223   V = { 50, 50},
7224   W = { 50, 50},
7225   X = { 50, 50},
7226   Y = { 50,  },
7227   Z = { 50,100},
7228   "00 = {250,250},
7229   "18 = {200,200},
7230   "3A = {200,150},
7231   "40 = {  ,100},
7232   "5E = {100,100},
7233   "5F = {100,100},
7234   "66 = { 50,  },
7235   "67 = {  , 50},
7236   "6E = {200,200}
7237 }
7238
7239 \SetProtrusion
7240 [ name      = euscript-vm,
7241   load      = euscript ]
7242 { encoding = U,
7243   family   = zeus  }
7244 {
7245   "01 = {600,600},
7246   "02 = {200,200},
7247   "03 = {200,200},
7248   "04 = {200,200},
7249   "05 = {150,150},
7250   "06 = {200,200},
7251   "07 = {200,200},
7252   "08 = {100,100},
7253   "09 = {100,100},
7254   "0A = {100,100},
7255   "0B = {100,100},
7256   "0C = {100,100},
7257   "0D = {100,100},
7258   "0E = {150,150},
7259   "0F = {100,100},
7260   "10 = {150,150},
7261   "11 = {100,100},
7262   "12 = {150,100},
7263   "13 = {100,150},
7264   "14 = {150,100},
7265   "15 = {100,150},
7266   "16 = {200,100},
7267   "17 = {100,200},

```

```

7268      "19 = {150,150},
7269      "1A = {150,100},
7270      "1B = {100,150},
7271      "1C = {100,100},
7272      "1D = {100,100},
7273      "1E = {250,100},
7274      "1F = {100,250},
7275      "20 = {150,200},
7276      "21 = {150,200},
7277      "22 = {150,150},
7278      "23 = {150,150},
7279      "24 = {100,200},
7280      "25 = {150,150},
7281      "26 = {150,150},
7282      "27 = {100,100},
7283      "28 = {100,100},
7284      "29 = {100,150},
7285      "2A = {100,100},
7286      "2B = {100,100},
7287      "2C = {100,100},
7288      "2D = {150,150},
7289      "2E = {150,150},
7290      "2F = {100,100},
7291      "30 = {100,100},
7292      "31 = {100,100},
7293      "32 = {100,100},
7294      "33 = {100,100},
7295      "34 = {100,100},
7296      "35 = {100,100},
7297      "3E = {150,150},
7298      "3F = {150,150},
7299      "60 = {    ,200},
7300      "61 = {200,   },
7301      "62 = {100,100},
7302      "63 = {100,100},
7303      "64 = {100,100},
7304      "65 = {100,100},
7305      "68 = {300,   },
7306      "69 = {    ,300},
7307      "6C = {100,100},
7308      "6D = {100,100},
7309      "6F = {100,100},
7310      "72 = {100,100},
7311      "73 = {200,100},
7312      "76 = {    ,100},
7313      "77 = {100,   },
7314      "78 = { 50, 50},
7315      "79 = {100,100},
7316      "7A = {100,100},
7317      "7D = {150,150},
7318      "7E = {100,100},
7319      "A8 = {100,100},
7320      "A9 = {100,100},
7321      "AB = {200,200},
7322      "BA = {    ,200},
7323      "BB = {    ,200},
7324      "BD = {200,200},
7325      "DE = {200,200}
7326    }
7327
7328    </eus>

Euler Fraktur font (eufrak).

7329    <*euf>
7330    \SetProtrusion

```

```

7331 [ name      = mathfrak ]
7332 { encoding = U,
7333   family   = euf  }
7334 {
7335   A = {   , 50},
7336   B = {   , 50},
7337   C = { 50, 50},
7338   D = {   , 80},
7339   E = { 50,   },
7340   G = {   , 50},
7341   L = {   , 80},
7342   O = {   , 50},
7343   T = {   , 80},
7344   X = { 80, 50},
7345   Z = { 80, 50},
7346   b = {   , 50},
7347   c = {   , 50},
7348   k = {   , 50},
7349   p = {   , 50},
7350   q = { 50,   },
7351   v = {   , 50},
7352   w = {   , 50},
7353   x = {   , 50},
7354   1 = {100,100},
7355   2 = { 80, 80},
7356   3 = { 80, 50},
7357   4 = { 80, 50},
7358   7 = { 50, 50},
7359   "12 = {500,500},
7360   "13 = {500,500},
7361   ! = {   ,200},
7362   ' = {200,300},
7363   ( = {200,   },
7364   ) = {   ,200},
7365   * = {200,200},
7366   + = {200,250},
7367   - = {200,200},
7368   {,} = {300,300},
7369   . = {400,400},
7370   {=} = {200,200},
7371   : = {   ,200},
7372   ; = {   ,200},
7373   ] = {   ,200}
7374 }
7375
7376 </euf>
7377 </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).

```

7378 <*cfg-e>
7379 \SetProtrusion
7380 <zpeu|euroitc> { encoding = U,
7381 <mvs> { encoding = {OT1,U},
7382 <zpeu>   family   = zpeu }
7383 <euroitc>   family   = {euroitc,euroitcs} }
7384 <mvs>   family   = mvs }
7385 {
7386 <zpeu>   E = {50, }
7387 <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

```

7388 <mvs>      164 = {50,50},    % \EUR
7389 <mvs>      068 = {50,-100}   % \EURdig
7390      }
7391
7392 <*zpeu|euroitc>
7393 \SetProtrusion
7394 { encoding = U,
7395   <zpeu>      family   = zpeu,
7396   <euroitc>   family   = {euroitc,euroitcs},
7397   shape       = it* }
7398 {
7399   <zpeu>      E = {100,-50}
7400   <euroitc>   E = {100,}
7401 }
7402
7403 </zpeu|euroitc>
7404 <*zpeu>
7405 \SetProtrusion
7406 { encoding = U,
7407   family   = {zpeus,eurosans} }
7408 {
7409   E = {100,50}
7410 }
7411
7412 \SetProtrusion
7413 { encoding = U,
7414   family   = {zpeus,eurosans},
7415   shape    = it* }
7416 {
7417   E = {200, }
7418 }
7419
7420 </zpeu>
7421 </cfg-e>

```

15.9 Interword spacing

Default unit is space.

```

7422 <*m-t>
7423 %%% -----
7424 %%% INTERWORD SPACING
7425
7426 \SetExtraSpacing
7427 [ name = default ]
7428 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
7429 {

```

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

7430 {,} = { , -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)

7431 r = { , -300, 300 },

- [before or] after lowercase characters with ascenders

7432 b = { , -200, 200 },

7433 d = { , -200, 200 },

7434 f = { , -200, 200 },

7435 h = { , -200, 200 },

7436 k = { , -200, 200 },

7437 l = { , -200, 200 },

7438 t = { , -200, 200 },

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., 'v', or 'w'

7439 c = { , -100, 100 },

7440 p = { , -100, 100 },

7441 v = { , -100, 100 },

7442 w = { , -100, 100 },

7443 z = { , -100, 100 },

7444 x = { , -100, 100 },

7445 y = { , -100, 100 },

- [before or] after lowercase characters with x-height plus descender without additional optical space

7446 i = { , 50, -50 },

7447 m = { , 50, -50 },

7448 n = { , 50, -50 },

7449 u = { , 50, -50 },

- after colon and semicolon

7450 : = { , 200, -200 },

7451 ; = { , 200, -200 },

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

7452 . = { , 250, -250 },

7453 ! = { , 250, -250 },

7454 ? = { , 250, -250 }

The order has to be reversed when enlarging is needed.'

7455 }

7456

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 pdf_{TeX} 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

```
7457 <*/m-t>
7458 \SetExtraSpacing
7459   [ name      = T2A,
7460     load      = default ]
7461   { encoding = T2A,
7462     family   = cmr }
7463   {
7464     \cyrg = { , -300, 300 },
7465     \cyrb = { , -200, 200 },
7466     \cyrk = { , -200, 200 },
7467     \cyrs = { , -100, 100 },
7468     \cyrr = { , -100, 100 },
7469     \cyrh = { , -100, 100 },
7470     \cyru = { , -100, 100 },
7471     \cyrt = { , 50, -50 },
7472     \cyrp = { , 50, -50 },
7473     \cyri = { , 50, -50 },
7474     \cyrishrt = { , 50, -50 },
7475   }
7476
7477 </m-t>
```

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.

```
7478 \SetExtraSpacing
7479   [ name      = nonfrench-cmr,
7480     load      = default,
7481     context   = nonfrench ]
7482   { encoding = {OT1,T1,Ly1,OT4,QX,T5},
7483     family   = cmr }
7484   {
```

²⁵ Contributed by *Karl Karlsson*.

latex.ltx has:

```
\def\nonfrenchspacing{
  \sfcode\~. 3000
  \sfcode\? 3000
  \sfcode\! 3000
```

```
7485   . = {333,2000,-667},
7486   ? = {333,2000,-667},
7487   ! = {333,2000,-667},
```

```
\sfcode\: 2000
```

```
7488   : = {333,1000,-500},
```

```
\sfcode\; 1500
```

```
7489   ; = {   , 500,-333},
```

```
\sfcode\{, 1250
```

```
7490   {,}= {   , 250,-200}
```

```
}
```

```
7491   }
7492
```

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.

```
7493 \SetExtraSpacing
7494   [ name      = nonfrench-default,
7495     load      = default,
7496     context   = nonfrench ]
7497   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
7498   {
7499     . = {240,2000,-667},
7500     ? = {240,2000,-667},
7501     ! = {240,2000,-667},
7502     : = {240,1000,-500},
7503     ; = {   , 500,-333},
7504     {,}= {   , 250,-200}
7505   }
7506
```

15.10 Additional kerning

Default unit is 1 em.

```
7507 %%% -----
7508 %%% ADDITIONAL KERNING
7509
```

A dummy list to be loaded when no context is active.

```
7510 \SetExtraKerning
7511   [ name = empty ]
7512   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
7513   { }
7514
```

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.)

```
7515 \SetExtraKerning
7516   [ name      = french-default,
7517     context    = french,
7518     unit       = space ]
7519   { encoding = {OT1,T1,LY1} }
7520   {
7521     : = {1000,}, % = \fontdimen2
7522     ; = {500, }, % ~ \thinspace
7523     ! = {500, },
7524     ? = {500, }
7525   }
7526
```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```
7527 \SetExtraKerning
7528   [ name      = french-guillemets,
7529     context    = french-guillemets,
7530     load       = french-default,
7531     unit       = space ]
7532   { encoding = {T1,LY1} }
7533   {
7534     \guillemotleft = { ,800}, % = 0.8\fontdimen2
7535     \guillemotright = {800, }
7536   }
7537
7538 \SetExtraKerning
7539   [ name      = french-guillemets-OT1,
7540     context    = french-guillemets,
7541     load       = french-default,
7542     unit       = space ]
7543   { encoding = OT1 }
7544   { }
7545
```

15.10.2 Turkish

```
7546 \SetExtraKerning
7547   [ name      = turkish,
7548     context    = turkish ]
7549   { encoding = {OT1,T1,LY1} }
7550   {
7551     : = {167, }, % = \thinspace
7552     ! = {167, },
7553     {=} = {167, }
7554   }
7555
7556 </m-t>
7557 </config>
```

26 http://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

16 OpenType font configuration files

The configuration files for OpenType fonts are not included in the documentation. Please refer to the respective `mt-.cfg` files in the microtype \TeX input directory.

17 Auxiliary file for micro fine tuning

This file can be used to test protrusion and expansion settings.

```

7558 (*test)
7559 \documentclass{article}
7560
7561 %% Here you can specify the font you want to test, using
7562 %% the commands \fontfamily, \fontseries and \fontshape.
7563 %% Make sure to end all lines with a comment character!
7564 \newcommand*{\TestFont}{%
7565   \fontfamily{ppl}%
7566   \fontseries{b}%
7567   \fontshape{it}% sc, sl
7568 }
7569
7570 \usepackage{ifthen}
7571 \usepackage[T1]{fontenc}
7572 \usepackage[latin1]{inputenc}
7573 \usepackage[verbose,expansion=alltext,stretch=50]{microtype}
7574
7575 \pagestyle{empty}
7576 \setlength{\parindent}{0pt}
7577 \newcommand*{\crulefill}{\cleaders\hbox{$\mkern-2mu\smash-\mkern-2mu$}\hfill}
7578 \newcommand*{\testprotrusion[2][]{%
7579   \ifthenelse{\equal{#1}{r}}{\crulefill}{\leftarrowfill} #2%
7580   \ifthenelse{\equal{#1}{l}}{\crulefill}{\rightarrowfill}
7581   \ifthenelse{\equal{#1}{r}}{\crulefill}{\leftarrowfill} #2
7582   \ifthenelse{\equal{#1}{l}}{\crulefill}{\rightarrowfill}
7583   you know the rest%
7584   \ifthenelse{\equal{#1}{l}}{\crulefill}{\rightarrowfill} #2%
7585   \linebreak
7586   {\fontencoding{\encodingdefault}%
7587   \fontseries{\seriesdefault}%
7588   \fontshape{\shapedefault}%
7589   \selectfont
7590   Here is the beginning of a line, \dotfill and here is its end}\linebreak
7591 }
7592 \newcommand*{\showTestFont}{\expandafter\stripprefix\meaning\TestFont}
7593 \def\stripprefix#1>{}
7594 \newcount\charcount
7595 \begin{document}
7596
7597 \microtypesetup{expansion=false}
7598
7599 {\centering The font in this document is called by:\\
7600 \texttt{\showTestFont}\par}\bigskip
7601
7602 \TestFont\selectfont
7603 This line intentionally left empty\linebreak
7604 %% A -- Z
7605 \charcount=65
7606 \loop
7607   \testprotrusion{\char\charcount}
7608   \advance\charcount 1
7609   \ifnum\charcount < 91 \repeat
7610 %% a -- z

```

```

7611 \charcount=97
7612 \loop
7613   \testprotrusion{\char\charcount}
7614   \advance\charcount 1
7615   \ifnum\charcount < 123 \repeat
7616 %% 0 -- 9
7617 \charcount=48
7618 \loop
7619   \testprotrusion{\char\charcount}
7620   \advance\charcount 1
7621   \ifnum\charcount < 58 \repeat
7622 %%
7623   \testprotrusion[r]{,}
7624   \testprotrusion[r]{.}
7625   \testprotrusion[r]{;}
7626   \testprotrusion[r]{:}
7627   \testprotrusion[r]{?}
7628   \testprotrusion[r]{!}
7629   \testprotrusion[l]{\textexclamdown}
7630   \testprotrusion[l]{\textquestiondown}
7631   \testprotrusion[r]{}}
7632   \testprotrusion[l]{({}
7633   \testprotrusion{/}
7634   \testprotrusion{\char`\}
7635   \testprotrusion{-}
7636   \testprotrusion{\textendash}
7637   \testprotrusion{\textemdash}
7638   \testprotrusion{\textquotelleft}
7639   \testprotrusion{\textquoteright}
7640   \testprotrusion{\textquotedblleft}
7641   \testprotrusion{\textquotedblright}
7642   \testprotrusion{\quotesinglbase}
7643   \testprotrusion{\quotedblbase}
7644   \testprotrusion{\guilsinglleft}
7645   \testprotrusion{\guilsinglright}
7646   \testprotrusion{\guillemotleft}
7647   \testprotrusion{\guillemotright}
7648
7649 \newpage
7650 The following displays the current font stretched by 5%,
7651 normal, and shrunk by 5%:
7652
7653 \bigskip
7654 \newlength{\MTln}
7655 \newcommand*{\teststring}
7656   {ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789}
7657 \settowidth{\MTln}{\teststring}
7658 \microtypesetup{expansion=true}
7659
7660 \parbox{1.05\MTln}{\teststring\linebreak\}
7661   \teststring\par\bigskip
7662 \parbox{0.95\MTln}{\teststring}
7663
7664 \end{document}
7665 /test

```

Needless to say that things may always be improved. For suggestions, mail to w.m.l@gmx.net.

A The title logo

7666 *`{*logo}`*

Here's how the logo on the title page was created.²⁷ 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.²⁸ 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`.

7667 `\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.

7668 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

7669 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

7670 `\newdimen\fboxrulei`

7671 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

7672 `\newdimen\fboxruleii`

7673 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

7674 `\newdimen\kernboxheight`

7675 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

7676 `\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).

7677 `\fontinstcc`

7678 `\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.

7679 `\ifdim\fontdimen6\font = 0pt`

7680 `\typeout{***Warning:~no-fontdimen-6-specified~***^^J%}`

7681 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi~***}`

7682 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi\relax`

7683 `\fi`

7684 `\installfonts`

7685 `\input_metrics{}\{\logofont,\metrics\printbbs{#1}\relax}`

7686 `\endinstallfonts`

7687 `}`

7688 `\normalcc`

Layers.

7689 `\makeatletter`

7690 `\def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}`

7691 `\ifx\mt@objects\undefined\let\mt@objects\@empty\fi`

7692 `\ifx\mt@order \undefined\let\mt@order \@empty\fi`

²⁷ 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.

²⁸ Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.ancor-online.net

```

7693 \xdef\mt@order{\mt@order[(Logo)]}
7694 \let\mtl@resources\empty
7695 \def\mtl@register#1{%
7696   \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
7697   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdfobj\space 0 R }
7698   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
7699   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
7700   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
7701 \mtl@register{canvas}
7702 \mtl@register{characters}
7703 \mtl@register{bounding-boxes}
7704 \mtl@register{TeX-boxes}
7705 \xdef\mt@order{\mt@order]}
7706 \global\let\mtl@objects\mt@objects
7707 \ifx\pdfcolorstack\undefined
7708   \pdfcatalog{/OCProperties <<
7709     /OCGs [\mt@objects]
7710     /D << /Order [\mt@order] >> >>}
7711 \fi
7712 \def\togglelayer#1#2{%
7713   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
7714   user{/Subtype/Link
7715     /BS << /Type/Border/W 0 >> /H/0
7716     /A << /S/SetOCGState
7717     /State[/Toggle \csname mtl@#1\endcsname] >>
7718   }#2\pdfendlink
7719 }

```

\printbbs Preparation.

```

7720 \setcommand\printbbs#1{%
7721   \setbox0\hbox{#1}%
7722   \leavevmode
7723   \kern-\fboxrulei

```

The canvas in the natural width of the text minus protrusion, in color bgcolor.

```

7724 \mtl@layer{canvas}{%
7725   \getboundarychars#1\relax
7726   \tempdim=\dimexpr\wd0 - (\scaletoem{\lcode\font\firstchar}+
7727     \scaletoem{\rcode\font\lastchar})\relax
7728   \kern\dimexpr\scaletoem{\lcode\font\firstchar}\relax
7729   \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
7730     \hrule width \tempdim
7731     height \dimexpr\dp0+\ht0+0.15em\relax}%
7732   \kern-\tempdim

```

The baseline, in color blcolor.

```

7733   \vbox{\color{blcolor}%
7734     \hrule width \tempdim
7735     height \fboxrulei}%
7736   }%
7737   \kern-\dimexpr\wd0 -\scaletoem{\rcode\font\lastchar}\relax

```

The string.

```

7738   \printbbs #1\relax\relax
7739 }

```

\getboundarychars Get first

```

7740 \def\getboundarychars#1#2\relax{%
7741   \def\firstchar{`#1}%
7742   \getlastchar#1#2\relax
7743 }

```

\getlastchar ... and last character.

```

7744 \def\getlastchar#1#2{%
7745   \ifx\relax#2\relax
7746     \def\lastchar{`#1}%
7747   \else
7748     \expandafter\getlastchar

```



```

7749 \fi
7750 #2%
7751 }

\printbss Loop over all characters of the string.
7752 \def\printbss#1#2#3\relax{%
7753 \ifx\relax#1\relax
7754 \else
7755 \ifx\relax#2\relax
7756 \printbb{#1}{}%
7757 \else
7758 \printbb{#1}{#2}%
7759 \fi
7760 \expandafter\printbss
7761 \fi
7762 #2#3\relax
7763 }

\printbb Record the kern between the current and the following character. \kerning is a fontinst
command.
7764 \setcommand\printbb#1#2{%
7765 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
7766 \showboxes{#1}%
This could be another application.
7767 % \quad
7768 % w: \the\scaletom{\width{#1}},
7769 % bb: \the\scaletom{\bbleft{#1}}/%
7770 % \the\scaletom{\bbright{#1}},
7771 % \the\scaletom{\number\numexpr\width{#1}-\bbright{#1}\relax}
7772 % h: \height{#1}/\bbtop{#1}, \bbbottom{#1}/\depth{#1}\par
7773 }

\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’
≠ ‘,‘).
7774 \setcommand\showboxes#1{%
7775 \leavevmode
7776 \color{texcolor}%
We have to record the width of the glyph.
7777 \setbox0\hbox{\color{texcolor}{#1}}%
7778 \global\tempdim=\wd0\relax
7779 \kern-\fboxrulei
1. The  $\TeX$  box: Print a frame in color texcolor. This frame shows the glyph as  $\TeX$  sees it.
7780 \mtl@layer{TeX-boxes}{%
7781 \hbox{%
7782 \lower\dimexpr \dp0 + \fboxrulei\relax
7783 \hbox{%
7784 \vbox{%
7785 \hrule height\fboxrulei
7786 \hbox{%
7787 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
7788 \phantom{\unhcopy0}%
7789 \vrule width\fboxrulei
7790 }%
7791 \hrule height\fboxrulei}}}%
7792 }%
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.
7793 \kern-\wd0
7794 \mtl@layer{characters}{%
7795 \hbox{\box0}%
7796 }%
Step back by the amount that the character’s bounding box differs from the  $\TeX$  box on the left side.
7797 \kern\dimexpr\scaletom{\bbleft{#1}}-\tempdim-\fboxruleii\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

7798 \mtl@layer{bounding-boxes}{%
7799   {\color{bbcolor}%
7800    \hbox{%
7801     \lower\dimexpr-\scaletom{\bbbottom{#1}}+\fboxruleii\relax
7802     \hbox{%
7803      \vbox{%
7804       \hrule height\fboxruleii
7805       \hbox to \dimexpr\scaletom{\numexpr
7806        \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
7807        \vrule height \dimexpr\scaletom{\numexpr
7808         \bbtop{#1}-\bbbottom{#1}\relax}%
7809         width\fboxruleii
7810         \hfill
7811         \vrule width\fboxruleii}%
7812       \hrule height\fboxruleii}}}%
7813     }%
7814     \kern-\dimexpr\fboxruleii+\fboxrulei\relax
7815   }%

```

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.

```

7816 \kern\scaletom{\numexpr\width{#1}-\bbright{#1}\relax}%
7817 \mtl@layer{TeX-boxes}{%
7818   {\ifnum\thekern<0
7819    \color{kerncolor}%
7820    \kern\scaletom{\thekern}%
7821    \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletom{\thekern}\relax
7822     height \kernboxheight}%
7823    \kern\scaletom{\thekern}%
7824   \else
7825    \color{texcolor}%
7826    \ifnum\thekern=0 \else
7827     \lower\kernboxheight
7828     \hbox{%
7829      \vbox{%
7830       \hrule height\fboxrulei
7831       \hbox{%
7832        \vrule height \kernboxheight width\fboxrulei
7833        \kern\dimexpr\scaletom{\thekern}-2\fboxrulei\relax
7834        \vrule width\fboxrulei
7835      }%
7836      \hrule height\fboxruleii}}%
7837     \fi
7838     \fi
7839   }%
7840   }%
7841   \kern-\fboxrulei
7842 }

```

```

7843 \newbox\logobox
7844 \def\printlogo{%
7845   \setbox\logobox=\hbox{\vbox{%
7846     \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

7847 \def\logofont{pkpri9e10}
7848 \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
7849 \font\thellogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

7850 %\def\logofont{pplri}
7851 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
7852 %\edef\logofont{\logofont8r}
7853 %\font\thellogofont=\logofont\space at 78pt

```

Load the font.

```

7854 \thelogo font
      Protrusion values (overdone for didactic reasons).
7855 \lcode\font`M=96
7856 \rcode\font`e=46

      Now we can generate the logo.
7857 \pdfliteral direct{/SXS gs}%
7858 \showlogo{Microtype}%
7859 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
7860 % \kern5pt\[\[3\baselineskip]
7861 % \long\def\@makefnmark##1{%
7862 % \leftskip 0pt
7863 % \parindent 0pt
7864 % \everypar{\parindent 0pt}%
7865 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
7866 % \footnotetext[1]{This graphic display on a
7867 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
7868 % their \togglelayer{bounding-boxes}{bounding boxes}
7869 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.}
7870 }%
7871 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
7872 \immediate\pdfobj{<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>}%
7873 \immediate\pdfxform
7874 attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
7875 resources {/Properties <<\mtl@resources>>
7876 /ExtGState << /SXS \the\pdflastobj\space 0 R >>
7877 }
7878 \logobox
7879 % \vskip-2.5\baselineskip
7880 \leavevmode
7881 % \togglelayer{characters}{%
7882 % \pdfrefxform\pdflastxform
7883 % }%
7884 \pdfannot\logodimens{%
7885 /Subtype/Widget /FT/Btn /T(Logo)
7886 %/F 4 % why did I say this?
7887 /AP << /N \the\pdflastxform\space 0 R >>
7888 /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
7889 /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
7890 /D << /S/SetOCGState /State[/Toggle \csname mtl@bounding-boxes\endcsname] >>
7891 /U << /S/SetOCGState /State[/Toggle \csname mtl@TeX-boxes\endcsname] >>
7892 >>
7893 }%
7894 \vspace{3\baselineskip}
7895 }

      Our font.
7896 \pdfmapline{+pkpmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmri8a10.pfb}

      Define colours (thered and thegreen are copied from microtype.dtx).
7897 \def\mtdefinecolors{
7898 \definecolor{thered}{rgb}{0.65,0.04,0.07}
7899 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
7900 \colorlet{texcolor}{thegreen!50} % TeX boxes
7901 \colorlet{kerncolor}{texcolor} % negative kerns
7902 \colorlet{bbcolor}{thered!50} % bounding box
7903 \colorlet{bgcolor}{black!8} % canvas
7904 \colorlet{blcolor}{black!50} % baseline
7905 \colorlet{textcolor}{black!40} % text
7906 }

      Use with microtype.dtx
7907 \ifx\documentclass\@twoclasseserror
7908 \usepackage{xcdraw}{xcolor}
7909 \mtdefinecolors
7910 \else

```

A.2 Document

Now we can start the document.

```

7911 \documentclass[10pt,a4paper]{ltxdoc}
7912 \providecommand\MakePercentComment{\relax}
7913 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

    Re-use the preamble from microtype.dtx.

7914 \usepackage{microtype-doc}
7915 \usepackage{attachfile}
7916 \makeatletter
7917 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
7918 \makeatother
7919 \begin{document}

    You are currently reading this.

7920 \DocInput{microtype-logo.dtx}

    And here's the logo.

7921 \vfill
7922 \begin{center}
7923   \printlogo \null
7924 \end{center}
7925 \vfill
7926 \expandafter\enddocument
7927 \fi

    That's it.

7928 /logo

```

B The letterspacing illustration

```

7929 \ifx\lssample\undefined
7930 {*lssample}

```

Upon popular request, here's how I've created the letterspacing illustration.²⁹

B.1 Macros

Rule width and image height and depth.

```

7931 \makeatletter
7932 \newdimen\lsamount
7933 \newdimen\lsrule
7934 \lsrule=0.2pt
7935 \def\lsheight{8pt}
7936 \def\lsdepth{12pt}

    Our font (Adobe Caslon).

7937 \def\lsfont{\fontfamily{paca}\selectfont}

    Loop over all letters in <#2>, letterspacing them by <#1>.

7938 \def\dols#1#2{\lsamount=#1\relax \dols#2\enddols}
7939 \def\dolss#1#2\enddols{%
7940   \ifx\empty#2\empty\divide\lsamount 2\fi
7941   \ls{#1}%
7942   \ifx\empty#2\empty\else \dols#2\enddols \fi
7943 }

```

One tikz picture for each letter.

```

7944 \def\ls#1{%
7945   \begin{tikzpicture}[remember picture,line width=\lsrule]
7946     \tikzstyle{every node}=[inner sep=0pt]

```

²⁹ 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, may can use the `pdftk` tool.

The bounding box.

```
7947 \mts@layer{stuff}{%
7948 \node[draw=thegrey,
7949 fill=theshade,
7950 outer sep=\lsrule,
7951 anchor=base,
7952 font=\lsfont]{\phantom{#1}};
7953 }
```

The letter.

```
7954 \node[anchor=base,font=\lsfont](#1){#1};
```

Two auxiliary coordinates.

```
7955 \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
7956 \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
7957 \mts@layer{stuff}{%
```

Now draw the normal character width,

```
7958 \draw[color=thered!75,
7959 fill=thered!30,
7960 outer sep=\lsrule]
7961 (#1L) rectangle (#1R);
7962 \ifdim\lsamount>0pt
7963 \path (#1.base east) ++(+.5\lsamount,-6pt) coordinate (#1_ls);
7964 \path (#1R) ++(\lsamount+\lsrule,\lsdepth) coordinate (#1E);
```

and the letter space.

```
7965 \draw[color=thered,
7966 fill=thered!50,
7967 outer sep=\lsrule]
7968 (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
7969 \fi
7970 }
7971 \end{tikzpicture}%
7972 \ignorespaces
7973 }
```

Draw the interword space.

```
7974 \def\lssp#1#2#3#4{%
7975 \mts@layer{stuff}{%
7976 \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
7977 \tikzstyle{every draw}=[anchor=bottom]
7978 \coordinate(#1space) at (#2/2,\lsdepth/2);
7979 \coordinate(#1stretch) at (#2+#3/2,+0pt);
7980 \coordinate(#1shrink) at (#2-#4/2,+0pt);
7981 \draw[color=thegreen,fill=thegreen!50,use as bounding box]
7982 (0,0) rectangle ++(+#2,\lsdepth);
7983 \draw[color=thegreen,fill=thegreen!30]
7984 (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
7985 \draw[color=thegreen,fill=thegreen!50]
7986 (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
7987 \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]
7988 (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
7989 \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
7990 (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
7991 \end{tikzpicture}%
7992 }\ignorespaces
7993 }
```

Layers.

```
7994 \def\mts@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
7995 \def\mts@layer#1#2{\pdfliteral{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral{EMC EMC}}
7996 \ifx\mt@objects\undefined\let\mt@objects\empty\fi
7997 \ifx\mt@order\undefined\let\mt@order\empty\fi
7998 \xdef\mt@order{\mt@order[(Sheep)]}
7999 \let\mts@resources\empty
8000 \def\mts@register#1{%
```

```

8001 \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
8002 \expandafter\xdef\csname mts@#1\endcsname{\the\pdflastobj\space 0 R }
8003 \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
8004 \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
8005 \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
8006 \mts@register{stuff}
8007 \mts@register{tracking}
8008 \mts@register{ispace}
8009 \mts@register{ospace}
8010 \mts@register{istretch}
8011 \mts@register{ishrink}
8012 \mts@register{ostretch}
8013 \mts@register{oshrink}
8014 \mts@register{okern}
8015 \mts@register{ligature}
8016 \mts@register{_compatibility}
8017 \xdef\mt@order{\mt@order}

```

Anchor point for the arrow in the code.

```

8018 \newcommand\anchorarrow[1]{%
8019 \tikz[remember picture,overlay]\node(#1_c){};}

```

Add an arrow from code to image.

```

8020 \newcommand\add@arrow[5][left]{%
8021 \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
8022 \mtsx@layer{#3}{%
8023 \draw[->,thick,color=the#2](#4) to[bend #1] (#5);}%
8024 }

```

Toggle layer.

```

8025 \def\toggle@layer#1#2#3{%
8026 \pdfstartlink
8027 user{/Subtype/Link
8028 /BS << /Type/Border/W 0 >> /H/0
8029 % /BS << /Type/Border/W 1 /S/D /D[4 1] >>
8030 % /C[0.7 0.7 0.7] /H/0
8031 /Contents(Click to Toggle!)
8032 /A << /S/SetOCGState
8033 /State[/Toggle \csname mts@#1\endcsname] >>
8034 }%
8035 \rlap{#2}%
8036 {\fboxsep=0pt \fboxrule=0pt
8037 \mtsx@layer{stuff}{%
8038 \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
8039 \mtsx@layer{#1}{%
8040 \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}}%
8041 }%
8042 \pdfendlink
8043 }
8044 \newcommand\showarrow[2][ ]{%
8045 \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
8046 \toggle@layer{\@tempa}{\itshape #2}}

```

The environment for our illustration.

```

8047 \def\ls@sample#1{%
8048 \parskip 4pt \parindent 0pt
8049 \par
8050 \vskip4pt
8051 {\leftskip 15pt
8052 \mt@pseudo@marg{\color{theblue}Click on the image to show the kerns
8053 and spacings involved. Click on emphasised words in the text below
8054 to reveal the relation of image and code.}
8055 \mt@layer{_compatibility}{%
8056 \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
8057 \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
8058 \mt@pseudo@marg{\color{thered}%
8059 If you had a \acronym{PDF} viewer that understands

```

```

8060 \acronym{PDF}\,\smaller{1.5}, you could hide the arrows selectively.}}
8061 \vskip-\mt@unvdimen}%
8062 \vskip-4pt
8063 \setlength\fbboxsep{4pt}%
8064 \leavevmode
8065 \pdfstartlink
8066   user{/Subtype/Link
8067         /BS << /Type/Border/W 0 >> /H/0
8068         /A << /S/SetOCGState
8069             /State[/Toggle \mts@stuff] >>
8070       }%
8071   \fcolorbox{theframe}{theshade}%
8072   {\fontsize{34}{38}\selectfont #1}%
8073 \pdfendlink
8074 \par\medskip
8075 }%
8076 \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
8077 }

```

Now define the illustration to be used in the document.

```

8078 \def\lssample{%
8079   \ls@sample{%
8080     \dols{0pt}{Stop}
8081     \lssp{o}{0.45em}{0.25em}{0.15em}
8082     \dols{0.16em}{\st{ealing}\hskip-\dimexpr 0.08em+\lsrule\relax
8083     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
8084     \dols{0.16em}{sheep}
8085     \dols{0pt}{!}
8086   }%

```

Don't forget to add the arrows.

```

8087 \vspace{-\baselineskip}
8088 \add@arrow{red} {tracking}{lsamount_c.east}{a_ls}
8089 \add@arrow{red} {okern} {okernend_c.east}{p_ls}
8090 \add@arrow{green} {ospace} {ospace_c.east} {ospace}
8091 \add@arrow{green} {ispace} {ispace_c.center} {ispace}
8092 \add@arrow{green!75} {istretch} {istretch_c.east} {istretch.north}
8093 \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
8094 \add@arrow{green!75} {ostretch} {ostretch_c.east} {ostretch.north}
8095 \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
8096 \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
8097 }
8098 \fi

```

This is for use with microtype.dtx

```

8099 \ifx\documentclass\@twoclasseserror
8100   \usepackage{tikz}
8101 \else

```

B.2 Document

```

8102 \documentclass[10pt,a4paper]{ltxdoc}
8103 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

```

Re-use the preamble from microtype.dtx.

```

8104 \usepackage{microtype-doc}
8105 \usepackage{attachfile}
8106 \usepackage{tikz}
8107 \makeatletter
8108 \pdfcatalog{/OCProperties <<
8109             /OCGs [\mt@objects]
8110             /D << /Order [\mt@order] /BaseState/OFF >>
8111             >> }
8112 \makeatother
8113 \begin{document}

```

You are currently reading this.

```

8114 \DocInput{microtype-lssample.dtx}
      Now show what we are able to do.
8115 \noindent
8116 Since a picture is worth a thousand words, probably even more if, in our
8117 case, it depicts a couple of letterspaced words, let's bring one to sum up
8118 these somewhat confusing options. Suppose you had the following settings
8119 (which I would in no way recommend; they are only for illustrative purposes):
8120 \begin{verbatim}
8121 \SetTracking
8122 [ no ligatures = {"\anchorarrow{nolig}"f},
8123   spacing      = {60"\anchorarrow{ispace}"0*,"%
8124                  "-1"\anchorarrow{istretch}"00*,"%\anchorarrow{ishrink}"},
8125   outer spacing = {4"\anchorarrow{ospace}"50,"%
8126                  "2"\anchorarrow{ostretch}"50,1"\anchorarrow{oshrink}"50},
8127   outer kerning = {"\anchorarrow{okernbegin}"*,"%
8128                  "\anchorarrow{okernend}"*} ]
8129 { encoding = * }
8130 { 1"\anchorarrow{lsamount}"60 }
8131 \end{verbatim}
8132 and then write:
8133 \begin{verbatim}
8134 Stop \textls{stealing sheep}!
8135 \end{verbatim}
8136 this is the (typographically dubious) outcome:
8137
8138 \lssample
8139
8140 \noindent
8141 While the word 'Stop' is not letterspaced, the space between the letters in
8142 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
8143 of 160/1000\,em\,=\allowbreak\,0.16\,em.
8144 The \showarrow[ispace]{inner-space}{green} within the letterspaced text is
8145 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
8146 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
8147 untouched.
8148 The \showarrow[ospace]{outer-space}{green} (of 0.45\,em) immediately before the
8149 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
8150 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
8151 Note that there is no outer space after the text, since the exclamation mark
8152 immediately follows; instead, the default \showarrow[okern]{outer-kern}{red}
8153 of half the letterspace amount (0.08\,em) is added.
8154 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
8155 neglected to specify the `|s|' in the |no ligatures| key.
8156
8157 \expandafter\enddocument
8158 \fi
8159 </lssample>

```


C Change history

2004/09/11 Version 1.0

General: Initial version 1

2004/09/21 Version 1.1

General: configuration file names in lowercase (suggested by *Harald Harders*) 81
 remove 8-bit characters from the configuration files (suggested by *Harald Harders*) 136
 Protrusion: add factors for some more characters 143
 settings for Adobe Minion (contributed by *Harald Harders*) 144
`\DeclareCharacterInheritance`: new command: possibility to specify character inheritance 111
`\MT@declare@sets`: remove spaces around set name 98
`\MT@DeclareSet`: remove spaces around first argument 97
`\MT@find@file`: fix: also check whether the file for the base font family has already been loaded 81
`\MT@get@basefamily`: only remove suffix if it is 'x' or 'j' 82
`\MT@get@listname@`: don't check for empty attributes list 83
`\MT@ifempty`: fix: use category code 12 for the percent character (reported by *Tom Kink*) 43
`\MT@is@number`: numbers may also be specified in hexadecimal or octal (suggested by *Harald Harders*) 88
`\MT@pdftex@no`: fix: version check (reported by *Harald Harders*) 38
`\MT@permute`: don't use sets for empty encoding 113
`\MT@setup@expansion`: issue an error instead of a warning, when pdf_T_EX version is too old for `autoexpand` 127
`\MT@split@codes`: fix: allow zero and negative values 60
`\MT@use@set`: remove spaces around set name 102
`\UseMicrotypeSet`: remove spaces around first argument 101

2004/10/03 Version 1.2

Font sets: declare `cmor` as an alias of `cmr` 134
 new: `allmath` and `basicmath` 133
 Protrusion: add settings for Computer Modern Roman and Adobe Garamond in T₁ encoding 167
 add settings for Computer Modern Roman math symbols 172
`\MT@familyalias`: define alias font name as an alternative, not as a replacement 56
`\MT@get@basefamily`: also remove 'w' (swash capitals) 83
`\MT@get@highlevel`: check whether defaults have changed 98
`\MT@get@inh@list`: fix: set inheritance list globally to `\@empty` 85
`\MT@get@listname@`: alternatively check for alias font name 83
`\MT@get@size`: additional magic to catch some errors hijack `\set@fontsize` instead of `\@setfontsize` 100
`\MT@loop`: fix: new macro, used instead of `\loop` 47
`\MT@maybe@do`: also check for alias font name 57
`\MT@permute@@@@`: more sanity checks for `\SetProtrusion` and `\SetExpansion` 114
`\MT@setupfont`: also search for alias font file 54
 fix: call `\@enc@update` if necessary 54

2004/10/27 Version 1.3

General: fix: specifying `load` option does no longer require to give a name, too 108
 Font sets: declare `aer`, `zer` and `hfor` as aliases of `cmr` 134
`\MT@fix@catcode`: check some category codes (compatibility with german) 34
`\MT@load@list`: check whether list exists 81

2004/11/12 Version 1.4

General: check for `pdfcprot` 52
 don't use scratch registers in global definitions 85
 use `\pickup@font` instead of `\define@newfont` as the hook for `\MT@setupfont` 92
 use one instead of five counters 48
 Protrusion: tweak quote characters for `cmr` variants (OT₁, T₁, lmr) 149
`\microtypesetup`: fix: set the correct levels, and remember them; warning when enabling an option disabled in package options 122
`\SetExpansion`: fix: specifying extra options does no longer require to give a name, too 105

2004/11/17 Version 1.4a

General: new option: `final` 119
`\MT@cfg@catcodes`: fix: reset some more catcodes when reading files (reported by *Michael Hoppe*) 82

2004/11/26 **Version 1.4b**

General: fix: set catcodes before reading global configuration file (reported by <i>Christoph Bier</i>)	121	form abcz (reported by <i>Georg Verwey</i>)	83
optimisation: use less <code>\expandafers</code> and <code>\csnames</code>	42	<code>\MT@get@slot</code> : don't define <code>\MT@char</code> globally (save stack problem)	85
Protrusion: harmonise dashes in upshape and italic (<code>cmr</code> , <code>pad</code> , <code>pp1</code>)	143	<code>\MT@ifdimen</code> : don't set <code>\MT@count</code> globally (save stack problem)	44
slanted like italics	152	<code>\MT@setup@PDF</code> : new message if <code>\pdfoutput</code> is changed	125
<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	102
<code>\MT@get@basefamily</code> : fix: failed for font names of the			

2004/12/15 **Version 1.5**

General: defaults: step: 4 (suggested by <i>Hàn Thế Thành</i>)	119	<code>\MT@get@highlevel</code> : don't test defaults if called after begin document	98
new option: selected, by default false (suggested by <i>Hàn Thế Thành</i>)	117	<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>	137	<code>\MT@set@pr@codes</code> : adjust protrusion factors before setting the inheriting characters	59
Protrusion: settings for Bitstream Charter	144	<code>\MT@setup@expansion</code> : defaults: calculate step as $\min(\text{stretch}, \text{shrink})/5$	126
<code>\DeclareMicrotypeAlias</code> : remove spaces around arguments	103	defaults: turn off expansion for DVI output	126
<code>\MT@cfg@catcodes</code> : reset catcode of '=' (compatibility with Turkish babel)	82	disable automatic expansion for DVI output	127
<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	120	tune CMR math letters (OML encoding)	172
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	119	<code>\MT@get@inh@list</code> : correct message if selected is false	85
restructure dtx file	133	<code>\MT@set@ex@codes</code> : introduce factor option	66
test whether <code>\pickup@font</code> has changed	94	<code>\MT@set@pr@codes</code> : introduce factor option	59
test whether numeric options receive a number	119	<code>\MT@setup@expansion</code> : disable automatic expansion for old pdfTeX versions	127
use e-TeX's <code>\ifcsname</code> and <code>\ifdefined</code> if defined	43	<code>\MT@use@set</code> : retain current set if new set is undeclared	102
Protrusion: add italic uppercase Greek letters	152	<code>\MT@vinfo</code> : new macro instead of <code>\ifMT@verbose</code>	35
improve settings for numbers (pointed out by <i>Peter Muthesius</i>)	145		

2005/02/02 **Version 1.6a**

Documentation: add table of fonts with tailored protrusion settings	21	reported by <i>Bernard Gaulle</i>)	85
<code>\MT@get@slot</code> : completely redone, hopefully more robust (compatible with frenchpro; 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>)	99	<code>\textbackslash</code> to T1 encoding	147
disallow automatic expansion if pdfTeX too old	111	<code>\DeclareMicrotypeAlias</code> : may also be used inside configuration files	103
fix: remove space after <code>autoexpand</code>	111	<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i>)	103
new value for verbose option: errors	119	<code>\Microtype@Hook</code> : new command for font package authors	121
shorter command names	48	<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	122
warning when running in draft mode	125	<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	82	sion	90
\MT@cfg@catcodes: reset catcode of ‘:’ (compatibility with french* packages)	82	\MT@scale: new macro: use e-TeX’s \numexpr if available	48
\MT@get@listname@: use \@tfor (<i>Andreas Böhmann’s</i> idea)	83	\MT@set@ex@codes: two versions of this macro	66
\MT@get@slot: remove backslash hack	85	\MT@setup@expansion: modify \showhyphens	128
test for \chardefed commands	86	\MT@split@name: don’t define \MT@encoding &c. \globally	56
test whether \(\encoding)\{...} is defined	86	\MT@test@ast: make it simpler	98
\MT@if@list@exists: don’t define \MT@#1@c@name \globally, here and elsewhere	84	\MT@try@order: always check for size, too (suggested by <i>Andreas Böhmann</i>)	83
\MT@ifdimen: comparison with 1 to allow size smaller than 1 (suggested by <i>Andreas Böhmann</i>)	44	fix: also check for //(\series)/(\shape)// (reported by <i>Andreas Böhmann</i>)	83
\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	110	\MT@find@file: no longer wrap names in commands	81
if font substitution has occurred, set up the substitute font, not the selected one	92	\MT@get@charwd: warning for missing (resp. zero-width) characters	61
new option: config to load a different main configuration file	121	\MT@get@font@dimen@six: new macro: test whether \fontdimen 6 is defined	59
new option: unit, by default character	120	\MT@get@listname@: made recursive	83
Documentation: add example for factor option	13	\MT@get@slot: fix: expand active characters	85
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	86
add hint about error messages	27	\MT@get@unit: new macro: get unit for codes	64
Font sets: add U encoding to allmath	133	\MT@in@rlist: made recursive	47
declare pxx and txx as aliases of ppl resp. ptm	135	\MT@is@active: new macro: translate inputencoding defined characters	89
Inheritance: remove \DJ from T1 list (it’s the same as \DH)	137	\MT@is@letter: warning for non-ASCII characters	88
Protrusion: add LY1 characters for Times	152	\MT@ledmac@setup: character protrusion with ledmac	50
settings for AMS math fonts	176	\MT@map@clist@n: new macro: used instead of \@for	46
verified settings for slanted Computer Modern Roman	160	\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>)	94	\MT@old@cmd: renamed commands from \..MicroType.. to \..Microtype..	35
\DeclareMicrotypeAlias: warning when overriding an alias font	103	\MT@pdf@tex@no: case 5: pdfTeX 1.30	38
\DeclareMicrotypeSetDefault: new command: set default font set	102	\MT@permute@@@@@: add ranges to the beginning of the lists	114
\MT@cfg@catcodes: reset catcodes of the remaining ASCII characters	82	\MT@scale: fix: remove spaces in e-TeX variant (reported by <i>Mark Rossi</i>)	48
\MT@check@rlist: made recursive	115	\MT@setupfont@hook: restore % and \# when hyperref is loaded	51
\MT@curr@list@name: new macro: current list type and name	91	restore csquotes’s active characters	51
\MT@declare@sets: warning when redefining a set	98	restore percent character if Spanish babel is loaded	51
\MT@define@set@key@: use comma lists instead of token lists	98	\MT@split@codes: get character width once only	60
		\MT@use@set: fix: remove braces in first line	102
		\MT@xadd: simplified	46

2005/10/28 Version 1.9

General: \DeclareMicrotypeSet: new key: font	100	Documentation: add hint about verbatim environment	25
\SetProtrusion: value ‘relative’ renamed to ‘character’ for key unit	110	add remark about Type 1 fonts required for automatic font expansion	8
allow context-specific font setup	92	Font sets: add OT4 encoding to text sets	133
compatibility with TeX Live hack (reported by <i>Herbert Voß</i>)	37	add T5 encoding to text sets	133
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	117	Inheritance: add list for OT4	138
option unit: rename value relative to character	120	add list for T5 (requested by <i>Hàn Thế Thành</i>)	139

Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR	147	<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)	143	<code>\MT@is@active</code> : redone: use <code>\set@display@protect</code>	89
settings for T5 encoded Computer Modern Roman	143	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code>	88
<code>\DisableLigatures</code> : new command: disable ligatures (requires pdf \TeX 1.30)	104	<code>\MT@maybe@do</code> : redone	57
<code>\microtypecontext</code> : new command: change setup context in the document	96	<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 \mathcal{E} - \TeX version	43	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero	128
<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code>	43	warning if user requested zero step	126
		<code>\MT@toks</code> : use instead of <code>\toks@</code>	40
		<code>\SetProtrusion</code> : (et al.) new key: font	104

2005/12/05 **Version 1.9a**

General: ‘ <code>{file name}/(line number)</code> ’ as default list name	108	diately (requested by <i>Georg Verweyen</i>)	98
new option: <code>defersetup</code> , by default true	118	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed	98
remove superfluous test whether <code>\pickup@font</code> has changed	94	<code>\MT@ifdefined@c@T</code> : new macros: true case only	43
Documentation: add explanation for error message in DVl 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> (mathdesign) as an alias of Charter	135	<code>\MT@in@clist</code> : fix	46
Protrusion: fix: remove ‘_’ from OT1 encoding	148	<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	143	<code>\MT@is@feature</code> : new macro: check for pdf \TeX feature	49
<code>\microtypesetup</code> : inside the preamble, accepts all package options	122	<code>\MT@map@clist@n</code> : following \LaTeX 3	46
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	95	<code>\MT@permute@ccc@</code> : don’t define permutations for unused encodings	114
<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	107
Protrusion: settings for URW Garamond	144	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b)	107

2006/05/05 **Version 1.9d**

Font sets: <code>md*</code> instead of <code>m</code> series in basic sets	133	<code>\DeclareCharacterInheritance</code> : fix: empty context	111
add QX encoding to text sets	133	<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>)	138	<code>\MT@get@ex@opt</code> : fix: evaluate preset	67
Protrusion: settings for QX encoding (contributed by <i>Maciej Eder</i>)	150	<code>\MT@get@font@dimen</code> : warning for zero fontdimen	62
settings for Euro symbols (Adobe, ITC, marvosym)	183	<code>\MT@get@opt</code> : optimise: don’t reset when preset option is set	64
tweak AMS settings	176	set list name before presetting	64

<code>\MT@is@active</code> : support for Unicode (inputenc/utf8) . . .	89	mandatory argument	104
<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	104
<code>\SetProtrusion</code> : (et al.) optimise: unify keys for			

2006/07/28 **Version 1.9e**

General: fix: default value for <code>activate</code> : true . . .	117	settings for Euler Roman font	179
Documentation: add hint about unknown encodings include LPPL	26	<code>\DeclareCharacterInheritance</code> : new key ‘inputenc’ to set the input encoding	111
Font sets: declare <code>zeur</code> and <code>zeus</code> (eulervm) as aliases of <code>eur</code> resp. <code>eus</code> (euler)	216	<code>\MT@rem@from@clist</code> : model after <code>\@removeelement</code>	47
Inheritance: adapt to marvosym’s changed encoding	135	<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	140	<code>\pickup@font</code> : no tracing with trace package	94
fix: forgotten comma in <code>mt-mvs.cfg</code> ; adapt to marvosym’s changed encoding	182	<code>\SetExpansion</code> : new key: inputenc	105
	183	<code>\SetProtrusion</code> : (et al.) new key: inputenc	104

2006/09/09 **Version 1.9f**

Protrusion: fix: euler-vm did not load euler settings . . .	180	<code>\MT@reset@context</code> : only reset context if it has actually been changed	96
<code>\MT@curr@list@name</code> : fix: <code>\MessageBreak</code> must not be expanded	91	<code>\MT@set@inh@list</code> : fix: forgotten comma in the features list	112
<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	106
<code>\MT@get@inh@list</code> : fix: input encoding must be set after the inheritance list has been parsed	85	<code>\SetProtrusion</code> : (et al.) set catcodes before parsing optional argument	104
<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>)	117	Font sets: add <code>footnotesize</code> and <code>scriptsize</code> sets	134
(beta:1) new option: <code>letterspace</code> , by default 100	119	add <code>smallcaps</code> set	134
(beta:3) compatibility with <code>soul</code> : register <code>\textls</code> and <code>\lsstyle</code>	53	<code>\DeclareMicrotypeBabelHook</code> : (beta:1) new command: interaction with <code>babel</code>	104
(beta:8) option ‘babel’: fix: switch off French <code>babel</code> ’s shorthands properly (reported by <i>Daniel Flipo</i>)	132	<code>\lsstyle</code> : (beta:1) new command: <code>letterspacing</code>	74
(beta:8) option ‘babel’: switch off Turkish <code>babel</code> ’s shorthands	132	(beta:8) fix: font switches don’t pose a problem anymore	74
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	74
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>	74
option ‘unit’, <code>\SetProtrusion</code> : deprecate value ‘relative’ completely	110	<code>\MT@declare@sets</code> : fix: empty size list when redefining set	98
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	89
add hint about warning when tracking and 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 ‘draft’ 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	131
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	106
		<code>\SetExtraSpacing</code> : (beta:1) new command: adjustment of interword spacing	106
		<code>\SetTracking</code> : new command: tracking	105
		<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	93	<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)	90
new option: <code>copyfonts</code>	118	<code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code>	72
simplify key declarations	108	<code>\MT@lua</code> : (basic) support for <code>LuaTeX</code>	39
Documentation: add hint about error message with <code>pdfTeX 1.40</code>	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@pdfTeX@no</code> : case 7: <code>pdfTeX 1.40.4</code>	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	135	<code>\MT@set@inputenc@</code> : only load <code>inputenc</code> files if necessary	64
declare <code>chr (chmath)</code> as an alias of Charter (reported by <i>Geoff Vallis</i>)	135	<code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in <code>pdfTeX 1.40.4</code>)	72
default set for tracking: <code>smallcaps</code>	134	possibility to customise interword spacing	72
Inheritance: remove <code>'- → '127'</code>	137	<code>\MT@setup@expansion</code> : warning if stretch or shrink aren't multiples of step	128
Protrusion: settings for Bitstream Letter Gothic	144	<code>\MT@setupfont</code> : don't call <code>\@@enc@update</code> anymore	54
Spacing: add sample	184	only add features that are available with the respective <code>pdfTeX</code>	54
Tracking: add ligatures that are to be disabled	141	<code>\MT@setupfont@hook</code> : restore percent character if Galician <code>babel</code> is loaded	51
<code>\DeclareMicrotypeVariants</code> : new command	103	<code>\MT@the@pr@code@tr</code> : adjust protrusion of letter-spaced fonts	60
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only	104	<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	130
<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	93
<code>\MT@get@charwd</code> : subtract letterspacing amount from width	62	<code>\SetTracking</code> : new key 'no ligatures' to disable ligatures of letterspaced fonts	105
<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>)	105
<code>\MT@ifdimen</code> : employ <code>LuaTeX</code> features if available	44	third argument may be empty	105
<code>\MT@ifint</code> : employ <code>LuaTeX</code> features if available	44	<code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code>	96
<code>\MT@ifstreq</code> : employ <code>LuaTeX</code> features if available	45		
fix: <code>ℒ-TeX</code> 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	90		

2007/12/23 **Version 2.3**

General: disable <code>\microtypecontext</code> in <code>hyperref's \pdfstringdef</code>	52	<i>phan Hennig</i>)	96
fix: really switch off Turkish shorthands	132	<code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code>	82
new value for verbose option: silent (suggested by <i>Karl Berry</i>)	119	<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	100
turned some warnings into errors	119	<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	89
add letterspacing illustration	17	<code>\MT@noligatures</code> : fix: set evaluation didn't work (bug introduced in 2.2)	80
<code>\do@subst@correction</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i>)	94	<code>\MT@plain</code> : letterspace: support for <code>eplain/minilts</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>Stephan Hennig</i>)			

<code>\MT@set@tr@codes</code> : also adjust tracking if protrusion is not enabled, and even for <code>letterspace</code> (reported by <i>Stephan Hennig</i>)	72	<code>\MT@SetTracking</code> : sanity check for value	105
possibility to customise outer kerning (suggested by <i>Stephan Hennig</i>)	72	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled	129
		<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 <code>letterspace</code> 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	125
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	149	<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	130		

2008/06/04 Version 2.3b

General: compatibility with <code>CJKutf8</code> : also check for its definition	93	known slots	81
<code>\MT@exp@gcs</code> : new macro: reduce save stack size	42	<code>\MT@requires@latex</code> : new macro	37
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i>)	55	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i>)	72
<code>\MT@get@size</code> : grouping	100	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode	77
<code>\MT@noligatures@</code> : fix: warning messages for unknown slots		make <code>\spaceskip-aware</code> (<code>ragged2e</code>)	77
		<code>\MT@tr@outer@r@</code> : additional test for horizontal mode	78

2008/11/11 Version 2.3c

General: <code>LuaTeX</code> supported by default	39	coding (reported by <i>Vasile Gaburici</i>)	139
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 <code>CJK</code>	26	<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>)	133	<code>\MT@setup@expansion</code> : default step: 1 for <code>pdfTeX</code> versions ≥ 1.40	126
<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 <code>LuaTeX</code>	74	fix: don't adjust in math mode (reported by <i>Christoph Bier</i>)	77
make invalid in math mode	74	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i>)	78
<code>\microtypesetup</code> : select font after setup	122	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning	80
<code>\MT@check@active@set</code> : warning for missing default sets	121	<code>\textls</code> : make math mode aware	79
<code>\MT@lua</code> : update for <code>LuaTeX</code> 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>)	186
Expansion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	141	<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@ls@aftergroup</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i>)	73
add T2A encoding	133	<code>\MT@setup@</code> : make space-unaware (requested by <i>Marcin Borkowski</i>)	49
Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	149	<code>\MT@tr@outer@r@</code> : fix: set current kerning and spacing again (found by <i>Lars Rönnbäck</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*) 149

2013/03/13 **Version 2.5**

General: allow contexts for \LaTeX 108
 disable ‘`DVIoutput`’ option for \XeTeX 118
 fix: check whether ‘`\langle file \rangle / \langle line \rangle`’ list name already exists (reported by *Till A. Heilmann*) 108
 letterspacing with \LaTeX 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 \LaTeX compatibility 26
 include `microtype-logo.dtx` and `microtype-lssample.dtx` 191
 Font sets: add EU1 and EU2 encodings 133
 declare `lmr` as alias of Latin Modern Roman (OpenType version) 134
 declare `lmsy` and `lmm` as aliases of `cmsy` resp. `cmm` (reported by *Jonas Hogstrom*) 134
 declare `zgm` etc. (`garamondx`) as aliases of URW Garamond 135
 declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version) 135
 Inheritance: add rudimentary list for EU1 and EU2 140
 Protrusion: add default lists for EU1 and EU2 ... 148
 improvements to Computer Modern Roman italics (contributed by *Hendrik Vogt*) 152
 Tracking: add EU2 encoding to default list 141
`\DeclareCharacterInheritance`: allow more than one encoding 111
`\DeclareMicrotypeAlias`: remove spaces in font name 103
`\ifMT@nofamily`: info if settings are not family-specific (suggested by *Hàn Thế Thành*) 59
`\LoadMicrotypeFile`: remove all spaces in font name 103

`\lstyle`: fix: ensure to set up math fonts (reported by *RazorXsr*) 74
`\microtypecontext`: fix: ensure to set up math fonts (reported by *RazorXsr*) 96
`\MT@define@code@key@family`: compatibility with `fontspec`: remove its internal counter (reported by *Till A. Heilmann*) 107
`\MT@define@code@key@font`: scrub `fontspec` feature count (found by *Meho R*) 107
`\MT@do@font`: adapt for \XeTeX 48
 adapt for \LaTeX 48
`\MT@get@slot@`: adapt for \XeTeX 86
 adapt for \LaTeX (requested by *Georg Duffner*) .. 86
`\MT@if@outer@next`: fix: conflict with `amsmath` (reported by *Scott Pakin*) 77
`\MT@info@missing@char`: fix error message for \XeTeX (reported by *Juan Acevedo*) 62
`\MT@is@charx`: compatibility with `xunicode` 90
`\MT@ledmac@setup`: fix to work with \XeTeX (reported by *Maïeul Rouquette*) 50
`\MT@ls@set@ls`: allow formulas in optional argument to `\textls` (fix by *Heiko Oberdiek*) 79
`\MT@register@subst@font`: only register if it isn’t registered already (reported by *George Gratzner* and *Josep Maria Font*) 95
`\MT@register@subst@font@cx`: only register if it isn’t registered already 95
`\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*) 129
`\MT@setupfont@hook`: restore `\%` and `\#` when `mathastext` is loaded (found by *Seamus Bradley*) 51
 select font with `fontspec` (found by *Georg Duffner*) 51

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:

<code>DVIoutput</code> 8	<code>letterspace</code> 8
<code>activate</code> 6	<code>protrusion</code> 6
<code>auto</code> 7	<code>selected</code> 8
<code>babel</code> 9	<code>shrink</code> 8
<code>config</code> 9	<code>spacing</code> 6
<code>draft</code> 9	<code>step</code> 8
<code>expansion</code> 6	<code>stretch</code> 8
<code>factor</code> 7	<code>tracking</code> 6
<code>final</code> 9	<code>unit</code> 7
<code>kerning</code> 6	<code>verbose</code> 9

- User Commands:**
- \DeclareCharacterInheritance 20
 - \DeclareMicrotypeAlias 21
 - \DeclareMicrotypeBabelHook 23
 - \DeclareMicrotypeSet 10
 - \DeclareMicrotypeSet* 10
 - \DeclareMicrotypeSetDefault 12
 - \DeclareMicrotypeVariants 20
 - \DeclareMicrotypeVariants* 20
 - \DisableLigatures 24
 - \LoadMicrotypeFile 21
 - \lslig 23
 - \lsstyle 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) 100
 - activate (option) 6, 117, 205
 - \add@accent 2681
 - ae (package) 21, 134
- B**
- babel (option) 9, 23, 26, 30, 117, 205
 - babel (package) 2, 3, 5, 16, 23, 24, 26, 51, 82, 104, 131, 132, 135, 202, 203, 205–207
- C**
- chemsym (package) 202
 - chmath (package) 21, 135, 206
 - CJK (package) 26, 53, 93, 206, 207
 - CJKutf8 (package) 93, 207
 - cm-super (package) 8
 - color (package) 9, 125
 - combine (package) 133, 205
- D**
- \DeclareCharacterInheritance 20, 41, 3472
 - \DeclareMicrotypeAlias 21, 43, 71, 3066
 - \DeclareMicrotypeBabelHook 23, 45, 64, 3128
 - \DeclareMicrotypeSet 10, 32, 57, 59, 72, 2794
 - \DeclareMicrotypeSet* 10, 2794
 - \DeclareMicrotypeSetDefault 12, 34, 61, 3025
 - \DeclareMicrotypeVariants 20, 42, 58, 63, 3053
 - \DeclareMicrotypeVariants* 20
 - deferssetup (option) 51, 117, 118, 122, 204
 - \define@newfont 2621, 2629, 2633, 2641
 - \DisableLigatures 24, 40, 62, 3101
 - \do@subst@correction 2678
 - docstrip (package) 33
 - draft (option) 9, 117, 119, 205
 - dsfont (package) 59
 - DVIoutput (option) 8, 9, 117, 118, 202, 208
- E**
- eco (package) 21, 134
 - \efcode 1457, 1483, 1484, 1524, 1526
 - eplain (package) 24, 30, 33, 37, 50, 125, 206
 - eucal (package) 180
 - eufrak (package) 182
 - euler (package) 91, 179, 205
 - eulervm (package) 21, 135, 180, 205
 - euroitc (package) 183
 - europus (package) 183
 - eurosans (package) 183
 - expansion (option) 6, 11, 117, 126
- F**
- \f@family 2635, 2636
 - \f@size 2628, 2630, 2632, 2639, 2679
 - factor (option) 7, 13, 31, 119, 120, 202
 - fancyvrb (package) 26
 - final (option) 9, 32, 117, 119, 201
 - fix-cm (package) 56
 - \font 900, 1837
 - \font@name 111, 945, 947, 949, 951, 961, 962, 1716, 1721, 1755, 1761, 1770, 1773, 1823, 1837, 1849, 1851, 1856, 1858, 1861, 1864, 1867, 1902, 1957, 2083, 2085, 2621, 2629, 2638, 2641, 2662, 2664, 2666, 2680, 2693, 2694, 2711, 2716
 - \fontcharwd 1201, 1205, 1211
 - \fontdimen 1124, 1131, 1236, 1241, 1340, 1343, 1733, 1734, 1761, 1849, 1895, 1896, 1903, 1904, 1910, 1914, 1921, 1931, 1935, 1938, 1957, 2096, 2099
 - fontinst (package) 187, 191, 193
 - fontinstallationguide (package) 166
 - fontspec (package) 6, 22, 24, 26, 27, 29, 56, 99, 107, 134, 208
 - fourier (package) 59
 - french (package) 203
 - frenchpro (package) 90, 202, 206
 - \glb@currsiz 1826, 2742
 - graphics (package) 9, 125
 - hyperref (package) 9, 51, 52, 92, 125, 203, 206
- G**
- garamondx (package) 21, 135, 208
 - german (package) 34, 201
- H**
- hfoldsty (package) 21, 134
- I**
- IEEEtran (package) 92
 - \iffontchar 1246
 - \ifMT@auto 282, 4203, 4254
 - \ifMT@babel 282, 4330, 4454
 - \ifMT@do 994, 1057, 1692, 2115
 - \ifMT@document 323, 3583
 - \ifMT@draft 282, 4108
 - \ifMT@expansion 282, 4166, 4251, 4287
 - \ifMT@fontspec 785, 790, 818, 975, 2858, 3254, 3280
 - \ifMT@if 281, 795, 802, 825, 857, 2707, 3686, 4474, 4477
 - \ifMT@inannot 99

- `\ifMT@inlist@` ... 598, 643, 716, 741, 918, 1024,
 1041, 1050, 1068, 1086, 1690, 2168, 2179,
 2228, 2277, 2694, 2701, 2713, 2768, 2780, 3084
`\ifMT@kerning` 282, 4344, 4467
`\ifMT@nofamily` 1096, 1103
`\ifMT@noligatures` 282, 4420
`\ifMT@nonselected` 1433, 2290, 2298
`\ifMT@noreset` 2359, 2376, 2388, 2402, 2446
`\ifMT@opt@auto` 3693, 4206
`\ifMT@opt@DVI` 3693, 4133
J `jurabib` (package) 92
K `kerning` (option) 6, 11, 30, 117
`keyval` (package) 40, 60, 136, 206
`\knaccode` 1641, 1642, 1652, 1655, 1661
L `ledmac` (package) 28, 31, 50, 92, 203
`ledpar` (package) 28
`\leftmarginkern` 756
`letterspace` (option) 8, 17, 23, 24, 30, 44, 71, 119, 205
`letterspace` (package)
 1, 24, 30, 31, 33, 35, 37, 72–74, 205–207
`\letterspacefont` 1721
`lineno` (package) 50
`listings` (package) 26, 53, 65, 82, 204, 205, 208
M `marvosym` (package) ... 21, 31, 33, 140, 183, 204, 205
`mathastext` (package) 51, 208
`mathdesign` (package) 21, 135, 204
`memoir` (package) 22, 92
`\MicrotypeHook` 121, 3948
`\microtypecontext`
 22, 47, 846, 2737, 2748, 4115, 4437, 4440
`microtypecontext` (environment) 22
`\microtypesetup` 9, 46, 3952, 4114
`miniltx` (package) 24, 30, 33, 37, 50, 206
`minimal` (package) 54
`\MT@font` 384, 921,
 943, 1126, 1237, 1250, 1704, 2231, 2303, 2667
`\MT@abbr@ex` 692
`\MT@abbr@ex@c` 692
`\MT@abbr@ex@inh` 692
`\MT@abbr@kn` 692
`\MT@abbr@kn@c` 692
`\MT@abbr@kn@inh` 692
`\MT@abbr@nl` 692
`\MT@abbr@pr` 692
`\MT@abbr@pr@c` 692
`\MT@abbr@pr@inh` 692
`\MT@abbr@sp` 692
`\MT@abbr@sp@c` 692
`\MT@abbr@sp@inh` 692
`\MT@abbr@tr` 692
`\MT@abbr@tr@c` 692
`\MT@active@features`
 953, 2695, 2698, 2710, 2721, 2757, 2767,
 3109, 4146, 4252, 4300, 4313, 4345, 4429, 4430
`\MT@addto@annot` 99
`\MT@addto@setup` 732, 733,
 805, 1353, 1786, 2615, 2737, 2738, 3341,
 3359, 3806, 3953, 4107, 4390, 4428, 4453, 4486
`\MT@adjustspacing` 4013, 4014, 4253
`\MT@auto` 1444, 4202, 4216, 4227, 4233
`\MT@auto@` 1444, 1453, 1497
`\MT@autofalse` 284, 4214, 4226
`\MT@autotrue` 284, 3908
`\ifMT@opt@expansion` 3693, 4162, 4288
`\ifMT@protrusion` 282, 748, 772, 4145, 4303
`\ifMT@selected` 282, 4237, 4257
`\ifMT@spacing` 282, 4312, 4329
`\ifMT@tracking` 282, 4299, 4359
`\ifMT@xunicode` 785, 2529, 2567
`ifpdf` (package) 9, 93, 125
`\iftracingmicrotypeinpdfall` 105
`inputenc` (package) 14,
 26, 31, 53, 64, 65, 86, 89, 91, 203, 205, 206
`\knbcbcode` 1634, 1635, 1651, 1654, 1660
`\knbscode` 1555, 1556, 1579, 1583, 1589
`lmodern` (package) 8, 149
`\LoadMicrotypeFile` 21, 44, 74, 3080, 4113
`\lpcode` 1138, 1174, 1175, 1378, 1381, 1746
`\lslig` 23, 54, 1842, 1845
`\lsstyle` 23, 51, 848, 873, 878,
 1824, 2022, 2028, 2033, 2036, 2046, 2050, 4116
`luaotfload` (package) 27, 87
`luatextra` (package) 41
`\MT@babelfalse` 291
`\MT@babeltrue` 291
`\MT@begin@catcodes` 2169, 2170, 2210, 3088, 3104,
 3138, 3154, 3178, 3199, 3213, 3478, 3923, 3924
`\MT@cat` 1349, 2606, 2608
`\MT@cfg@catcodes` 859, 1371, 2195, 2212
`\MT@char` 1163, 1164, 1174,
 1175, 1180, 1181, 1184, 1186, 1201, 1202,
 1205, 1378–1382, 1483, 1484, 1486, 1487,
 1524–1526, 1555, 1556, 1562, 1563, 1569,
 1570, 1573, 1574, 1579–1584, 1634, 1635,
 1641, 1642, 1645, 1646, 1651–1655, 2126,
 2332, 2348, 2356, 2358, 2368, 2370, 2378,
 2379, 2387, 2388, 2393, 2395, 2401, 2403,
 2406, 2410, 2414, 2418, 2513, 2516, 2518,
 2562, 2565, 2568, 2570, 3528–3530, 3535–3537
`\MT@char@` 1207,
 1208, 1211, 1246, 2332, 2338, 2343, 2348,
 2396, 2450, 2452, 2458, 2459, 2461, 2474,
 2475, 2478, 2479, 2482, 2483, 2487, 2489,
 2517, 2536, 2539, 2542, 2551, 2553, 2566, 2580
`\MT@charstring` 2345, 2516, 2521
`\MT@charxstring` 2546, 2570
`\MT@check@active@set`
 3934, 4152, 4270, 4302, 4322, 4349
`\MT@check@font` 917, 2691, 2759
`\MT@check@font@cx` 2696, 2759
`\MT@check@range` 3657, 3659
`\MT@check@range@` 3659, 3660
`\MT@check@rlist` 3610, 3650
`\MT@check@rlist@` 3650, 3651
`\MT@check@step` 4258
`\MT@checklist@` 1002, 1017, 2112
`\MT@checklist@family` 1035
`\MT@checklist@font` 1079
`\MT@checklist@size` 1063
`\MT@checksetup` 3964, 3975, 3982, 4001, 4031
`\MT@clear@options` 161, 274, 727, 3897, 4100
`\MT@clist@break` 570, 1030, 1058, 1074, 1088
`\MT@clist@function` 570

- \MT@cnt@encoding 3549, 3557, 3558
 \MT@cnt@family 3555, 3564, 3565
 \MT@cnt@series 3562, 3571, 3572
 \MT@cnt@shape 3569, 3577, 3578
 \MT@config@file 3912, 3921, 3922, 3926, 3927, 3930
 \MT@context 2253, 2265, 2284, 2311
 \MT@copy@font 904,
 939, 3337, 3343, 3355, 3362, 3815, 3817, 4136
 \MT@copy@font@ .. 939, 3337, 3343, 3355, 3362, 3815
 \MT@count 489, 490,
 676, 1193, 1201, 1203, 1205, 1209, 1211,
 1214, 1220, 1225, 1226, 1230, 1241, 1272, 1334
 \MT@curr@file 2173, 2174, 2183, 2184,
 3090, 3091, 3311, 3313, 3316, 3500, 3926, 4488
 \MT@curr@list@name ... 1239, 1275, 1365, 1493,
 2123, 2154, 2322, 2575, 2583, 2588, 2596, 2602
 \MT@curr@ls 1752, 1846, 1849
 \MT@curr@set@name
 2819, 2821–2824, 2826, 2827, 2832, 2837,
 2841, 2842, 2870, 2874, 2903, 2933, 2939,
 2945, 2949, 2950, 3193, 3269, 3290, 3305,
 3324, 3396, 3400, 3407, 3410, 3412, 3418,
 3421, 3424, 3436, 3449, 3457, 3467, 3501, 3503
 \MT@declare@char@inh 3483, 3488, 3498
 \MT@declare@sets 2802, 2808, 2820, 3111
 \MT@DeclareSet 2797, 2799, 2816
 \MT@DeclareSetAndUseIt 2796, 2815
 \MT@DeclareVariants 3056, 3057, 3059
 \MT@def@bool@opt 3770,
 3783, 3784, 3787, 3801, 3813, 3823, 3835
 \MT@def@n 373, 2790, 2791
 \MT@default@ex@set 3041
 \MT@default@kn@set 3041
 \MT@default@pr@set 3041
 \MT@default@sp@set 3041
 \MT@default@tr@set 3041
 \MT@define@code@key
 .. 3237, 3297, 3299, 3300, 3510, 3512, 3513
 \MT@define@code@key@family 3248, 3298, 3511
 \MT@define@code@key@font 3274, 3302, 3515
 \MT@define@code@key@size 3262, 3301, 3514
 \MT@define@opt@key .. 3303, 3326–3329, 3463–3465
 \MT@define@optionX 3952, 4010, 4013
 \MT@define@optionX@ 4021, 4050, 4052, 4053
 \MT@define@set@key@ 2830, 2989–2992
 \MT@define@set@key@font 2937, 2994
 \MT@define@set@key@size 2863, 2993
 \MT@detokenize@c 414, 2340, 2514
 \MT@detokenize@n 414, 2564
 \MT@dimen@six
 .. 1123, 1194, 1230, 1272, 1930, 1933, 2092
 \MT@dinfo 87
 \MT@dinfo@list
 1015, 1025, 1028, 1033, 1042, 1045, 1051,
 1053, 1061, 1069, 1072, 1077, 1087, 1090, 1094
 \MT@dinfo@nl 87
 \MT@DisableLigatures 3101
 \MT@do@font 660, 1140, 1457, 1592, 1662, 1746
 \MT@documentfalse 323
 \MT@documenttrue 323, 4432
 \MT@dofalse 994, 1006, 1029, 1046, 1073, 1091
 \MT@dotrue .. 994, 997, 1026, 1043, 1052, 1070, 2107
 \MT@draftfalse 287, 3837
 \MT@drafttrue 287, 3839
 \MT@edef@n 375, 2289, 2315, 2784,
 3243, 3257, 3313, 3316, 3319, 3499, 3637, 3869
 \MT@encoding 972, 1083, 1104, 1106, 1109,
 2248, 2260, 2340, 2514, 2563, 2587, 2594, 2602
 \MT@end@catcodes 2171, 2214, 3096,
 3113, 3149, 3173, 3194, 3210, 3224, 3493, 3925
 \MT@endinput 165, 3892, 3901
 \MT@engine 173,
 255, 259, 265, 271, 2023, 2027, 3893, 3899
 \MT@engine@tooold 173, 256
 \MT@error 76, 719,
 808, 2023, 2034, 2146, 2159, 2901, 3020,
 3047, 3119, 3346, 3372, 3379, 3825, 3987,
 4005, 4039, 4207, 4220, 4289, 4360, 4373, 4449
 \MT@error@doesntwork 4356
 \MT@ex@c@name 1426,
 1428, 1499, 1500, 1510, 1516, 1517, 1529, 3157
 \MT@ex@context 943, 2751, 2789
 \MT@ex@doc@contexts 2789
 \MT@ex@factor 292, 1445, 1503
 \MT@ex@factor@ . 1445, 1459, 1463, 1473, 1474, 1497
 \MT@ex@inh@name 1485–1487
 \MT@ex@level 292, 4253, 4255
 \MT@ex@max 304, 1476, 1477
 \MT@ex@min 304, 1479, 1480
 \MT@ex@setname 3012
 \MT@ex@split@val 1471
 \MT@exp@cs 370, 373, 376,
 379, 382, 954, 956, 965, 1067, 1185, 1259,
 1262, 1292, 1296, 1306, 1309, 1312, 1487,
 1574, 1646, 2276, 2714, 2722, 3523, 2727,
 2770, 2781, 2869, 3268, 3288, 3537, 3612, 3655
 \MT@exp@gcs 370, 374, 378, 380, 383
 \MT@exp@one@n 385, 419, 577,
 715, 908, 910, 966, 1021, 1084, 1689, 2178,
 2691, 2693, 2699, 2711, 2733, 2767, 2778,
 2789, 2808, 2988, 3005, 3034, 3083, 3488, 4437
 \MT@exp@two@c 387, 405, 411,
 420, 612, 907, 947, 949, 951, 960, 1823,
 1858, 1861, 1864, 2339, 2344, 2515, 2568, 2569
 \MT@exp@two@n 389, 1039, 1048, 2836, 2944
 \MT@expandfont 1430, 1448, 1452
 \MT@expansion 923, 1417, 4280
 \MT@expansionfalse 283, 4163, 4247
 \MT@expansiontrue 283, 3907
 \MT@extra@context 2789, 3141, 3157,
 3182, 3202, 3216, 3283, 3286, 3287, 3289,
 3330, 3338, 3344, 3356, 3361, 3474, 3609,
 3613, 3616, 3619, 3622, 3626, 3627, 3629, 3656
 \MT@extra@inputenc 3472, 3497, 3502, 3503
 \MT@factor@default 316, 3876, 4149
 \MT@family 908, 972,
 1040, 1083, 1104, 1106, 1109, 2249, 3075, 3076
 \MT@familyalias
 .. 909, 910, 983, 1047, 1049, 2259, 2261, 3077
 \MT@feat 994, 1128, 1166, 1254, 1256,
 1258, 1259, 1261, 1262, 1268, 1270, 1273,
 1281–1285, 1287, 1289–1293, 1296, 1297,
 1302, 1306, 1309, 1312, 1315–1317, 1326,
 1327, 1332, 1350, 1827, 2144, 2146, 2149,
 2150, 2154, 2156, 2159, 2285, 2287, 2289,
 2293, 2297, 2301, 2302, 2305, 2313, 2315,
 2316, 2318, 2325, 2329, 2576, 2577, 2606, 2608
 \MT@features
 712, 2789, 2802, 2988, 2999, 3028, 3308, 3483

- \MT@features@long 712, 715, 720, 2764, 3495
 \MT@file@list 2165,
 2167, 2175, 2178, 2180, 2185, 2188, 3083, 3087
 \MT@find@file 908, 910, 2165
 \MT@fix@catcode 5
 \MT@fix@font@set 2855, 4120
 \MT@font 384, 672,
 790, 819, 907, 960, 967, 1124, 1131, 1138,
 1139, 1174, 1175, 1180, 1181, 1201, 1202,
 1205, 1208, 1211, 1236, 1241, 1246, 1340,
 1343, 1378, 1379, 1381, 1382, 1453, 1457,
 1483, 1484, 1524, 1526, 1555, 1556, 1562,
 1563, 1569, 1570, 1579–1581, 1583, 1584,
 1589–1591, 1634, 1635, 1641, 1642, 1651,
 1652, 1654, 1655, 1660, 1661, 1689, 1693,
 1751, 2116, 2355, 2409, 2610, 2662–2664,
 2666, 2679, 2680, 2691, 2692, 2699, 2725, 2734
 \MT@font@copy 943, 944, 951, 952, 960–962
 \MT@font@list 2610, 2691–2694, 2758
 \MT@font@orig 944
 \MT@font@sets 2840, 2855, 2948, 4120
 \MT@fontspecfalse 786
 \MT@fontspectrue 786, 788, 816
 \MT@gdef@n 373, 3132, 3148,
 3172, 3209, 3223, 3412, 3436, 3449, 3505, 3942
 \MT@get@axis 2962–2965, 2975
 \MT@get@basefamily 2177, 2215
 \MT@get@basefamily@ 2221, 2224
 \MT@get@char@unit 1165, 1304, 1330, 1422
 \MT@get@charwd 1199, 1230, 1307, 1330
 \MT@get@config 3912
 \MT@get@ex@opt 1421, 1439, 1497
 \MT@get@ex@opt@ 1505–1507, 1509, 1515
 \MT@get@font 2943, 2953
 \MT@get@font@ 2954, 2959, 3295
 \MT@get@font@and@size 3279, 3294
 \MT@get@font@dimen 1234, 1310
 \MT@get@font@dimen@six 1112, 1123, 1539, 1618, 1705
 \MT@get@highlevel ... 2835, 2845, 2977, 3242, 3253
 \MT@get@inh@list 1115, 1424, 1542, 1621, 2311
 \MT@get@listname 2230, 2287, 2313
 \MT@get@listname@ 2230
 \MT@get@ls@basefont 1720, 1848, 1855
 \MT@get@opt 1113, 1279, 1540, 1619
 \MT@get@range 2867, 2877, 3266
 \MT@get@size 2884, 2889, 2898, 2917, 2973
 \MT@get@slot 1162, 2125, 2332, 3527, 3534
 \MT@get@slot@ 2349, 2353
 \MT@get@space@unit
 .. 1304, 1406, 1553, 1560, 1567, 1632, 1639
 \MT@get@tr@opt 1707, 1794
 \MT@get@tr@opt@ 1811–1814, 1816
 \MT@get@unit 1312, 1320, 1804
 \MT@get@unit@ 1320
 \MT@getkey 4087, 4102
 \MT@glet 368, 380, 947, 1858,
 1889, 2028, 2036, 2660, 2759–2762, 3077,
 3337, 3355, 3552, 3815, 3817, 4401, 4406, 4492
 \MT@glet@nc 379, 1109, 1902, 2083, 2325,
 2680, 2758, 2824, 2826, 2832, 2939, 3165,
 3396, 3400, 3407, 3410, 3418, 3421, 3424, 3457
 \MT@glet@nn 382, 2777, 3939
 \MT@if@false 281,
 791, 798, 821, 838, 2704, 3654, 4468, 4475
 \MT@if@list@exists
 .. 1102, 1420, 1438, 1538, 1617, 1706, 2284
 \MT@if@outer@next
 ... 1973, 1988, 1992, 1995, 2000, 2005–2007
 \MT@if@true 281,
 792–794, 799–801, 822–824, 851, 855, 856,
 2697, 3663, 3667, 3675, 3680, 4469–4473, 4476
 \MT@ifdefined@c@T 393,
 795, 827, 1183, 1219, 1485, 1572, 1644,
 1739, 2502, 3075, 3502, 3599, 3948, 3951, 4495
 \MT@ifdefined@c@TF
 393, 749, 1710, 1723, 1726, 1737,
 1846, 1927, 2089, 2120, 2121, 2288, 2314, 3608
 \MT@ifdefined@n@T
 . 393, 738, 1018, 1036, 1064, 1080, 1184,
 1315, 1351, 1486, 1510, 1573, 1645, 1796,
 1798, 1817, 2267, 2605, 2822, 3070, 3160,
 3320, 3558, 3565, 3572, 3578, 3609, 3619, 4087
 \MT@ifdefined@n@TF 393, 983, 1000, 1019, 1037,
 1065, 1081, 1104, 1281, 1289, 1499, 1516,
 2110, 2149, 2255, 2340, 3014, 3017, 3043,
 3311, 3585, 3636, 3645, 3711, 3935, 3938, 4435
 \MT@ifdim .. 511, 632, 633, 637, 638, 2900, 2909,
 3661–3663, 3665, 3666, 3673–3675, 3678, 3679
 \MT@ifdimen 474, 2928, 3399, 3409, 3423, 3885
 \MT@ifempty 426, 1138, 1139, 1172, 1178,
 1395, 1396, 1552, 1559, 1566, 1589–1591,
 1603–1605, 1607–1609, 1631, 1638, 1660,
 1661, 1677, 1678, 1908, 1909, 1919, 1937,
 2058, 2081, 2082, 2094, 2769, 2801, 2806,
 2847, 2853, 2878, 2879, 2892, 2893, 2967,
 2978, 2998, 3003, 3027, 3032, 3108, 3189,
 3304, 3310, 3330, 3336, 3344, 3354, 3361,
 3434, 3482, 3487, 3724, 3755, 3915, 3961, 4028
 \MT@ifint 439, 2481, 3190, 3435, 3868, 3878
 \MT@ifstreq . 520, 1357, 2145, 2286, 2301, 2537,
 2540, 2552, 2732, 2942, 3278, 3398, 3420,
 3422, 3600, 3727, 3729, 3732, 3735, 3757,
 3759, 3773, 3774, 3845, 3846, 3851, 3855,
 3884, 3963, 3970, 3974, 3981, 4030, 4036, 4332
 \MT@in@clist 598,
 715, 739, 1021, 1039, 1048, 1084, 1689,
 2167, 2178, 2691, 2693, 2699, 2711, 2767, 3083
 \MT@in@rlist 626, 1067, 2276
 \MT@in@rlist@ 626
 \MT@in@rlist@@ 626
 \MT@in@tlist 614, 2227, 2778
 \MT@in@tlist@ 614
 \MT@inannotfalse 100
 \MT@inannottrue 100
 \MT@increment 676, 3557, 3564, 3571, 3577
 \MT@info 76, 91, 96, 3852, 4071, 4075
 \MT@info@missing@char 1214, 1244, 2405
 \MT@info@nl 76, 92, 93, 97, 750, 1105,
 1245, 3843, 3853, 3922, 3936, 3940, 4132,
 4136, 4148, 4155, 4254, 4281, 4301, 4306,
 4315, 4325, 4348, 4352, 4365, 4380, 4445, 4456
 \MT@inh@do 2324, 3516
 \MT@inh@feat 3472, 3482, 3485, 3496
 \MT@inh@split 3518, 3522
 \MT@inlist@false 598, 602, 615, 627, 2707
 \MT@inlist@true 598, 604, 622, 634, 639, 2707
 \MT@is@active 2339, 2494
 \MT@is@char 2344, 2515, 2521
 \MT@is@charx 2530, 2546, 2569

- \backslash MT@is@composite 2342, 2560
 \backslash MT@is@feature 714, 2807, 3004, 3033
 \backslash MT@is@letter 2337, 2438, 2518, 2565
 \backslash MT@is@number 2464, 2469
 \backslash MT@is@symbol 2341, 2512
 \backslash MT@iterate 648
 \backslash MT@kerning 927, 1614, 4351
 \backslash MT@kerningfalse 289
 \backslash MT@kerningtrue 289
 \backslash MT@kn@cc@name 1623, 1625, 1668, 3216
 \backslash MT@kn@context 2752, 2789
 \backslash MT@kn@doc@contexts 2789
 \backslash MT@kn@factor 292
 \backslash MT@kn@factor@ 1281
 \backslash MT@kn@inh@name 1644–1646
 \backslash MT@kn@max 304
 \backslash MT@kn@min 304
 \backslash MT@kn@setname 3012
 \backslash MT@kn@split@val 1629
 \backslash MT@kn@unit 298
 \backslash MT@kn@unit@ 1289, 1671
 \backslash MT@led@kern 750
 \backslash MT@led@unhbox@line 750
 \backslash MT@led@mac@setup 744, 814
 \backslash MT@let@cn 381, 382, 383, 984, 1119, 1428, 1500, 1546, 1625, 1797, 1799, 1803, 2144, 2156, 2281, 2285, 2318, 2662, 2921, 3324, 3501
 \backslash MT@let@nc .. 379, 1317, 2011, 2297, 2329, 2702, 3638, 3643, 3733, 3736, 3977, 3984, 4363, 4378
 \backslash MT@let@nn 382, 1282, 1287, 1290, 1302, 1517, 1520, 1818
 \backslash MT@letterspace 319, 1710, 1797, 1807, 4409, 4410, 4412
 \backslash MT@letterspace@ 1149, 1219, 1220, 1710, 1711, 1714, 1717, 1721, 1725, 1733, 1745–1747, 1752, 1761, 1933, 1937, 2044, 2059–2062, 2095, 2098
 \backslash MT@letterspace@default 319, 4410
 \backslash MT@listname 2230, 2278, 2281, 2288, 2289, 2291, 2293, 2314, 2315, 2317, 2318, 2322, 2325, 3537, 3542
 \backslash MT@listname@count 3307, 3312, 3314
 \backslash MT@load@inputenc 1358, 1361, 1370
 \backslash MT@load@list 1117, 1426, 1544, 1623, 2141
 \backslash MT@loop 648, 656, 670, 3642
 \backslash MT@lower 2871, 2877, 3269
 \backslash MT@ls@adjust 1759, 2055
 \backslash MT@ls@adjust@ 2039, 2063
 \backslash MT@ls@adjust@empty 2041, 2055
 \backslash MT@ls@adjust@relax 2042, 2055
 \backslash MT@ls@aftergroup 1765, 1778, 1783
 \backslash MT@ls@basefont 1855, 1864, 1867, 1868
 \backslash MT@ls@outer@k 1762, 1771, 1783, 1950, 1964, 2008, 2103
 \backslash MT@ls@set@ls 2046, 2049, 2055
 \backslash MT@ls@too@large 2062, 2065, 4412
 \backslash MT@lsfont 1716, 1721, 1733, 1734, 1743, 1746, 1747, 1751, 1823, 1833, 1878, 1880, 1895, 1896, 1903, 1904, 1910, 1914, 1921, 1931, 1935, 1938, 2096, 2099
 \backslash MT@lslig 1845
 \backslash MT@lua 242, 339, 463, 499, 543
 \backslash MT@luatex@no 240, 332
 \backslash MT@map@clist@ 570, 953, 2124, 2698, 2710, 2721, 2757, 2764, 2802, 2804, 2999, 3001, 3028, 3030, 3308, 3483, 3485, 3495, 4085
 \backslash MT@map@clist@n 570, 999, 2109, 2789, 2833, 2865, 2940, 2988, 3060, 3130, 3240, 3251, 3264, 3276, 3416, 3432, 3531, 3719, 3722, 3751, 3753, 3782, 3863, 3959, 4026
 \backslash MT@map@tlist@ 588
 \backslash MT@map@tlist@c 588, 617, 628, 956, 1185, 1487, 1574, 1646, 2234, 2727, 3610, 3655, 4120
 \backslash MT@map@tlist@n 588, 2206, 2523, 3551
 \backslash MT@max@char 2438, 2461
 \backslash MT@max@slot 664, 2438, 2487
 \backslash MT@maybe@do ... 994, 1099, 1418, 1536, 1615, 1691
 \backslash MT@maybe@etex 149
 \backslash MT@maybe@rem@from@list 2727, 2731
 \backslash MT@MT 2, 76, 79, 82, 84, 86, 91, 92, 94, 95, 270, 726, 809, 811, 3346, 3895, 3916, 4071, 4075, 4133, 4336, 4449, 4450
 \backslash MT@next@listname 2237, 2246
 \backslash MT@next@listname@ 2256, 2268, 2274
 \backslash MT@n@ligatures 2116, 3101
 \backslash MT@n@l@setname 2108, 3101
 \backslash MT@nofamilyfalse 1096, 1101
 \backslash MT@nofamilytrue 1096, 2247
 \backslash MT@noligatures 931, 2105, 4421
 \backslash MT@noligatures@ 1878, 1880, 2116, 2119
 \backslash MT@noligaturesfalse 286
 \backslash MT@noligaturestrue 286, 3110
 \backslash MT@nonselectedfalse 1434, 1449
 \backslash MT@nonselectedtrue 1434, 1437
 \backslash MT@noresetfalse 2447, 2454, 2463, 2484, 2537, 2540, 2552
 \backslash MT@noresettrue 2335, 2447, 2472
 \backslash MT@old@cmd 66, 71–74
 \backslash MT@opt@autofalse 3694
 \backslash MT@opt@autotruer 3694, 3784
 \backslash MT@opt@def@set 3710, 3738, 3763
 \backslash MT@opt@DVIfalse 3695
 \backslash MT@opt@DVIttrue 3695, 3790, 3793
 \backslash MT@opt@expansionfalse 3693
 \backslash MT@opt@expansiontrue 3693
 \backslash MT@options 3227
 \backslash MT@optwarn@admissible 3696, 3775, 3855
 \backslash MT@optwarn@nan 3700, 3870, 3880
 \backslash MT@orig@add@accent 2681
 \backslash MT@orig@foreign@language 4462, 4464
 \backslash MT@orig@pickupfont 844, 887, 892, 2616, 2683
 \backslash MT@orig@py@macron 884, 888, 893
 \backslash MT@orig@select@language 4457, 4459
 \backslash MT@outer@kern ... 1761, 1764, 1769, 1771, 1772, 1777, 1849, 1850, 1852, 2008, 2009, 2076, 2103
 \backslash MT@outer@space 1755, 1756, 1893, 1946, 1948, 1949, 1962, 1963, 1980, 1981, 1996, 1997, 2001, 2002
 \backslash MT@pdf@annot 99
 \backslash MT@pdf@or@lua 178, 183, 268, 442
 \backslash MT@pdftex@no 199, 331
 \backslash MT@permute 3147, 3171, 3187, 3208, 3222, 3508, 3548
 \backslash MT@permute@ 3548
 \backslash MT@permute@@ 3548
 \backslash MT@permute@@@ 3548
 \backslash MT@permute@@@@ 3576, 3581
 \backslash MT@permute@@@@@ 3589, 3591

- \MT@permute@define 3582, 3592–3594, 3634
 \MT@permute@reset 3551, 3640
 \MT@permutelist
 3141, 3157, 3185, 3202, 3216, 3283,
 3284, 3289, 3506, 3604, 3605, 3609, 3613,
 3616, 3619–3622, 3629, 3630, 3656, 3687, 3688
 \MT@pickupfont 889, 894, 2677, 2685
 \MT@plain 125, 146
 \MT@pr@ccname 1117, 1119, 1386, 3141
 \MT@pr@context 943, 2751, 2789
 \MT@pr@doc@contexts 2789
 \MT@pr@factor 292, 3879, 4149, 4150
 \MT@pr@factor@ 1281
 \MT@pr@inh@name 1183, 1184, 1186
 \MT@pr@level 292, 4147, 4148
 \MT@pr@max 304
 \MT@pr@min 304
 \MT@pr@setname 3012
 \MT@pr@split@val 1170
 \MT@pr@unit 298, 3886, 4151
 \MT@pr@unit@ 1289, 1389
 \MT@preset@aux 1391,
 1393, 1395, 1396, 1399, 1673, 1675, 1677, 1678
 \MT@preset@aux@factor 1391, 1399, 1603–1605, 1673
 \MT@preset@aux@space . 1393, 1399, 1607–1609, 1675
 \MT@preset@ex 1511, 1528
 \MT@preset@kn 1666
 \MT@preset@kn@ 1666
 \MT@preset@pr 1384
 \MT@preset@pr@ 1384
 \MT@preset@sp 1596
 \MT@preset@sp@ 1596
 \MT@ProcessOptionsWithKV 4079, 4103
 \MT@protrudechars 4010, 4014, 4147, 4303
 \MT@protrusion 922, 1098, 4154
 \MT@protrusionfalse 282
 \MT@protrusiontrue 282, 3902
 \MT@rbba@expansion 707
 \MT@rbba@kerning 707
 \MT@rbba@protrusion 707
 \MT@rbba@spacing 707
 \MT@rbba@tracking 707
 \MT@register@font 935, 2692, 2760
 \MT@register@font@cx 2720, 2760
 \MT@register@subst@font 2668, 2693, 2761
 \MT@register@subst@font@cx 2709, 2761
 \MT@rem@from@clist 609, 966, 2733, 3232
 \MT@rem@from@list 956, 964
 \MT@rem@last@space 414
 \MT@repeat 648, 658, 672, 3648
 \MT@requires@latex 145, 162, 730, 840,
 871, 1822, 2614, 3702, 3862, 4080, 4084, 4491
 \MT@requires@luatex 326, 914, 1147, 1685, 4023, 4297
 \MT@requires@pdftex 326,
 441, 475, 521, 745, 903, 913, 926, 930,
 941, 1146, 1191, 1217, 1461, 1535, 1614,
 1684, 1875, 2105, 3102, 3334, 3448, 3456,
 3812, 3906, 4022, 4174, 4204, 4232, 4296, 4419
 \MT@res@a 533, 535, 600, 607, 610, 612, 616, 621
 \MT@res@b 534, 535, 610–612, 620, 621
 \MT@reset@context 2741, 2745, 2749, 2776
 \MT@reset@context@ 2749, 2776
 \MT@reset@ef@codes 1423, 1447, 1460, 1512
 \MT@reset@ef@codes@ 1455, 1464, 1469
 \MT@reset@kn@codes 1620, 1627, 1657
 \MT@reset@kn@codes@ 1657
 \MT@reset@pr@codes 1114, 1121, 1142
 \MT@reset@pr@codes@ 1142
 \MT@reset@sp@codes 1541, 1548, 1586
 \MT@reset@sp@codes@ 1586
 \MT@restore@catcodes 5, 7, 8, 163, 4497
 \MT@restore@p@h 782, 802, 857
 \MT@saved@setupfont 4068, 4076
 \MT@scale 683, 1194, 1255, 1269, 1272, 1474
 \MT@scale@factor 1196, 1227, 1253, 1401, 1530
 \MT@scale@to@em .. 1173, 1179, 1191, 1224, 1407,
 1554, 1561, 1568, 1633, 1640, 1806, 1928, 2090
 \MT@scrubfeature 976, 987, 3255
 \MT@scrubfeatures 987, 2859, 3281
 \MT@selectedfalse 285
 \MT@selectedtrue 285
 \MT@series 972, 1083, 2250, 2262
 \MT@set@all@ex 1455, 1531
 \MT@set@all@kn 1657, 1679
 \MT@set@all@pr 1135, 1142, 1397
 \MT@set@all@sp 1586, 1611
 \MT@set@babel@context 4434, 4460, 4465, 4479
 \MT@set@codes . 1120, 1153, 1429, 1547, 1626, 2157
 \MT@set@curr@ls 1752
 \MT@set@curr@ok
 .. 1764, 1777, 1783, 1790, 1791, 1985, 1989
 \MT@set@curr@os 1754, 1790, 1791, 1985, 1989
 \MT@set@default@set 3028, 3034, 3041
 \MT@set@ex@codes 1451, 4238, 4240
 \MT@set@ex@codes@n 1433, 1451, 4240
 \MT@set@ex@codes@s 1419, 4238
 \MT@set@ex@heirs 1487, 1523
 \MT@set@inh@list 3479, 3481
 \MT@set@inputenc 1116, 1348, 1425, 1543, 1622, 2323
 \MT@set@inputenc@ 1351, 1353
 \MT@set@kn@codes 1616
 \MT@set@kn@heirs 1646, 1650
 \MT@set@listname 1118,
 1280, 1427, 1498, 1545, 1624, 1795, 2575
 \MT@set@lsbasefont 1864
 \MT@set@lsfont 1750, 1821, 1833
 \MT@set@named@keys 3143, 3159, 3183, 3204, 3218, 3227
 \MT@set@pr@codes 1100
 \MT@set@pr@heirs 1187, 1376
 \MT@set@sp@codes 1537
 \MT@set@sp@heirs 1574, 1578
 \MT@set@tr@codes 1701, 1828, 1838
 \MT@set@tr@zero 1712, 1864, 4050
 \MT@SetExpansion 3155, 3157
 \MT@SetExtraKerning 3214, 3216
 \MT@SetExtraSpacing 3200, 3202
 \MT@SetProtrusion 3139, 3141
 \MT@SetTracking 3179, 3181
 \MT@setup@ 728, 732, 3804, 3805, 4492
 \MT@setup@contexts 2740, 2756
 \MT@setup@copies 4119, 4135
 \MT@setup@expansion 4122, 4159
 \MT@setup@kerning 4126, 4343, 4386
 \MT@setup@noligatures 4127, 4417
 \MT@setup@PDF 4118, 4130, 4140
 \MT@setup@protrusion 4121, 4143
 \MT@setup@spacing 4125, 4311, 4387
 \MT@setup@spacing@check 4328, 4495, 4496
 \MT@setup@tracking 4123, 4295, 4358, 4385
 \MT@setup@warntracking 4124, 4390

- \MT@setupfont .. 901, 2671, 4069, 4072, 4076, 4112
 \MT@setupfont@hook
 789, 817, 819, 826, 831, 857, 864, 902
 \MT@shape 972, 1083, 2251, 2263
 \MT@shorthandoff 4443, 4474, 4477
 \MT@show@pdfannot 106
 \MT@shrink 301, 1442,
 4170, 4171, 4178, 4179, 4182, 4186, 4243, 4256
 \MT@shrink@ 1442, 1453, 1497
 \MT@shrink@default 317
 \MT@size 633, 637, 638, 972, 1083
 \MT@size@name 626, 2278
 \MT@sp@cc@name 1544, 1546, 1598, 3202
 \MT@sp@context 2752, 2789, 4332
 \MT@sp@doc@contexts 2789
 \MT@sp@factor 292
 \MT@sp@factor@ 1281
 \MT@sp@inh@name 1572–1574
 \MT@sp@max 304
 \MT@sp@min 304
 \MT@sp@setname 3012
 \MT@sp@split@val 1550
 \MT@sp@unit 298
 \MT@sp@unit@ 1289, 1601
 \MT@spacing 927, 1534, 4324
 \MT@spacingfalse 288
 \MT@spacingtrue 288
 \MT@split@codes 1155, 1159
 \MT@split@name 907, 972
 \MT@step 301, 1443, 4173,
 4175, 4193, 4197, 4200, 4257, 4260, 4261, 4265
 \MT@step@ 1443, 1453, 1497
 \MT@stretch 301, 1441, 4167,
 4168, 4171, 4178, 4180, 4185, 4188, 4242, 4256
 \MT@stretch@ 1441, 1453, 1497
 \MT@stretch@default 317, 4168
 \MT@strip@prefix 2530, 2546, 2568
 \MT@temp 1137–1140, 1588–
 1592, 1659–1662, 1898, 1902, 1905, 1910,
 1913, 1918, 1977, 1980, 1986, 1990, 1993,
 1996, 2001, 2008, 2014, 2080, 2083, 2100,
 2961, 2986, 4083, 4088, 4093, 4094, 4096, 4099
 \MT@tempencoding 3584, 3585, 3595, 3601
 \MT@tempfamily 3596
 \MT@tempseries 3597
 \MT@tempshape 3598
 \MT@tempsize 3552, 3599, 3608, 3610, 3614
 \MT@test@ast 1926, 2088, 2846, 2851
 \MT@textls 49, 2041, 2042, 2044
 \MT@the@pr@code 1144, 1174, 1180, 1748
 \MT@the@pr@code@tr 1144, 1748
 \MT@tlist@break 588, 623, 645, 2239, 3690
 \MT@toks 280, 1245, 1275, 1493, 2336, 2371, 2397,
 2452, 2459, 2506, 2542, 2553, 2580, 2592, 2600
 \MT@tr@cc@name
 ... 1796–1799, 1803, 1817, 1818, 3184, 3191
 \MT@tr@context 2752, 2789
 \MT@tr@doc@contexts 2789
 \MT@tr@factor@ 1805
 \MT@tr@font@list 1687
 \MT@tr@ispace 1723, 1724, 1811
 \MT@tr@ligatures 1739, 1814, 1877, 1880
 \MT@tr@max 304, 2066–2068
 \MT@tr@min 304, 2070–2072
 \MT@tr@noligatures 1739, 1873
 \MT@tr@okern 1737, 1813
 \MT@tr@ospace 1726, 1727, 1811
 \MT@tr@outer@icr 1990, 2016
 \MT@tr@outer@icr@ 2016
 \MT@tr@outer@l 1757, 1942
 \MT@tr@outer@next 1970, 1974, 2018
 \MT@tr@outer@r . 1790, 1791, 1970, 1986, 1993, 2016
 \MT@tr@outer@r@ 1971, 1976
 \MT@tr@set@okern 1738, 2076
 \MT@tr@set@okern@ 2081, 2082, 2087
 \MT@tr@set@space 1729, 1894
 \MT@tr@set@space@ 1899–1901, 1907
 \MT@tr@set@space@@ 1912, 1917, 1920, 1925
 \MT@tr@setname 3012
 \MT@tr@unit@ 1798, 1927, 2089
 \MT@tracking 915, 1687, 1828, 2673, 4050, 4051, 4305
 \MT@tracking@ 1687, 4051
 \MT@trackingfalse 290, 4362
 \MT@trackingtrue 290
 \MT@try@order 2234, 2242
 \MT@undefined@char 2499, 2501, 2511
 \MT@upper 2871, 2877, 3269
 \MT@use@set 2999, 3005, 3012
 \MT@val 2769–2771, 2773, 2779,
 2781, 2784, 2834, 2838, 2846, 2848, 2866–
 2868, 2880, 2883, 2885, 2888, 2890, 2891,
 2897, 2899, 2900, 2902, 2905, 2907, 2918,
 2921, 2925, 2928–2930, 2932, 2934, 2941–
 2943, 2946, 2955–2957, 2969, 2971, 2976,
 2978, 2980, 2982, 3241, 3243, 3252, 3255,
 3257, 3265–3267, 3277–3279, 3285, 3290,
 3530, 3537, 3541, 3542, 3711, 3712, 3715,
 3723, 3724, 3727, 3729, 3732, 3735, 3754,
 3755, 3757, 3759, 3960, 3961, 3963, 3970,
 3974, 3981, 3987, 4027, 4028, 4030, 4036, 4039
 \MT@variants 2227, 3053, 3063
 \MT@vinfo 76, 93, 921, 1012, 1284,
 1293, 1297, 1332, 1501, 1518, 1704, 1714,
 2127, 2136, 2150, 2174, 2184, 2187, 2291,
 2293, 2299, 2750, 2773, 3085, 3091, 3843,
 3850, 3966, 3972, 3978, 3985, 4033, 4038, 4436
 \MT@warn@ascii 2461, 2579
 \MT@warn@axis@empty 2968, 2979, 2984
 \MT@warn@code@too@large 1259, 1262, 1266
 \MT@warn@err 76, 3847, 3848
 \MT@warn@ex@too@large 1477, 1480, 1491
 \MT@warn@maybe@inputenc ... 2122, 2593, 2601, 2604
 \MT@warn@nodim 3388, 3401, 3411, 3425
 \MT@warn@number@too@large 2488, 2585
 \MT@warn@preset@twidth ... 1390, 1410, 1602, 1672
 \MT@warn@rest 2377, 2413, 2590
 \MT@warn@tracking@DVI 1715, 4392
 \MT@warn@unknown ... 2367, 2383, 2392, 2417, 2598
 \MT@warning 67, 76, 94, 2067, 2071,
 2130, 2302, 2823, 2932, 2985, 3071, 3093,
 3192, 3321, 3390, 3438, 3452, 3602, 3620,
 3687, 3847, 3852, 3948, 4019, 4058, 4060, 4061
 \MT@warning@nl 76, 88, 95, 257,
 725, 762, 773, 833, 1125, 1237, 1273, 1325,
 1365, 1411, 1492, 1886, 2580, 2586, 2591,
 2599, 2644, 3162, 3363, 3697, 3704, 3715,
 3796, 3848, 3853, 3887, 3894, 3929, 3943,
 4109, 4198, 4244, 4263, 4317, 4333, 4396, 4481
 \MT@while@num 654, 662, 860–862

- `\MT@with@babel@and@T`
 ... 737, 792–794, 822–824, 4469–4473, 4476
`\MT@with@package@T`
 734, 787, 788, 796, 799–801, 807, 814–816,
 829, 841, 855, 856, 858, 872, 883, 2655, 4316
`\MT@xadd` 104, 554, 2175, 2180, 2185,
 2188, 2714, 2723, 2781, 2869, 3087, 3268, 3537
- N** `\normalfont` 900, 4273, 4277
- P** `\pdfadjustinterwordglue` 4052, 4314
`\pdfadjustspacing` 4015, 4276
`\pdfappendkern` 4054, 4347
`\pdfcopyfont` 951
`pdfcpot` (package) 26, 29, 52, 61, 92, 201
`\pdffontexpand` 1453
`\pdfnoligatures` 2129, 2135
`\pdfprependkern` 4053, 4346
`\pdfprotrudechars` 4014, 4405
`pdftexcmds` (package) 36, 207
- Q** `qfonts` (package) 21, 135, 203
- R** `ragged2e` (package) 26, 77, 129, 207, 208
`relsize` (package) 100
- S** `selected` (option) 8, 14, 32, 66, 84, 117, 202
`\selectfont` 1829, 2744, 2754, 3953, 4487
`\set@fontsize` 2925
`\SetExpansion` 14, 36, 3152
`\SetExtraKerning` 18, 38, 3212
`\SetExtraSpacing` 19, 39, 3197
`\SetProtrusion` 12, 35, 3136
`\SetTracking` 15, 37, 3176
`\sfcode` 4331
`\shbscode` 1569, 1570, 1581, 1584, 1591
`\showhyphens` 4271, 4275
- T** `\tagcode` 1743, 2121, 2126
`tex4ht` (package) 51, 205
`\textls` 23,
 52, 53, 849, 874, 879, 1760, 1825, 1841, 2039
`\textls*` 23
`\textmicrotypecontext` 22, 48, 845, 2748
`tikz` (package) 73, 196, 207
`tipa` (package) 92
- U** `ucs` (package) 89, 206
`ulgothic` (package) 21
- V** `verbose` (option) 9, 30, 31, 119, 202, 203, 206
- W** `wordcount` (package) 49, 206
- X** `\XeTeXcharglyph` 2403
`\XeTeXcountglyphs` 672
`\XeTeXfonttype` 2355
`\XeTeXglyph` 1208
`\XeTeXglyphindex` 2390
- Y** `yfonts` (package) 24
- Z** `zefonts` (package) 21, 134
- `\MT@xaddb` 562, 3288, 3612
`\MT@xdef@n` .. 375, 3015, 3018, 3045, 3050, 3074,
 3191, 3283, 3305, 3467, 3503, 3629, 3712, 3714
`\MT@xetex@no` 231
`\MT@xunicodefalse` 785
`\MT@xunicodetrue` 785, 787, 815
- `\nullfont` 900
`\pdftracingfonts` 335
`\pickup@font` 844, 887, 889,
 892, 894, 2643, 2645, 2654, 2677, 2683, 2685
`pifont` (package) 92
`pinyin` (package) 53, 206, 207
`protrusion` (option) 6, 11, 117
`PSNFSS` (package) 187
`pstricks` (package) 9, 125
`pxfonts` (package) 21, 135
- `\rightmarginkern` 755
`\rPCODE` 1139, 1180, 1181, 1379, 1382, 1747
`shrink` (option) .. 8, 15, 32, 119, 126, 128, 204, 206
`slantsc` (package) 166
`soul` (package) 4, 29, 53, 77, 205, 207
`soulutf8` (package) 29, 207
`\spacefactor` 1953
`\spaceskip` 1954, 1957
`spacing` (option) 6, 11, 26, 30, 117, 207, 208
`\stbscode` 1562, 1563, 1580, 1584, 1590
`step` (option) .. 8, 30, 32, 119, 126, 128, 202, 204, 206
`stretch` (option) ... 8, 15, 25, 32, 119, 126, 128, 206
- `trace` (package) 94, 205
`\tracingmicrotype` 87
`\tracingmicrotypeinpdf` 98
`\tracingmicrotypeinpdfallfalse` 105
`\tracingmicrotypeinpdfalltrue` 105
`tracking` (option) 6, 11, 23, 30, 117
`txfonts` (package) 21, 135
- `unit` (option) 7, 31, 203, 205
`\UseMicrotypeSet` 11, 33, 60, 73, 2817, 2996
- `\XeTeXglyphname` 2409
`\XeTeXprotrudechars` 4018
`\xspaceskip` 1956, 1959
`xunicode` (package) 29, 90, 91, 208

E The \LaTeX Project Public License

LPPL Version 1.3c 2006-05-20

Copyright 1999, 2002–2006 \LaTeX 3 Project

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

Preamble

The \LaTeX Project Public License (LPPL) is the primary license under which the the \LaTeX kernel and the base \LaTeX 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 \TeX -related (such as a \LaTeX package), but it is written in such a way that you can use it even if your work is unrelated to \TeX .

The section ‘WHETHER AND HOW TO DISTRIBUTE WORKS UNDER THIS LICENSE’, below, gives instructions, examples, and recommendations for authors who are considering distributing their works under this license.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the \LaTeX 3 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 ‘`cfgguide.tex`’ and ‘`modguide.tex`’ in the base \LaTeX 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 ‘ \LaTeX -Format’ or in the case of files belonging to the ‘ \LaTeX -format’ a program implementing the ‘ \TeX language’.

Conditions on Distribution and Modification

- Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
- You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.
- You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
- If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Main-

tainer are considered to be updated versions of the Work.

5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
 - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
 - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
 - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
 - (d) You distribute at least one of the following with the Derived Work:
 - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
 - ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.
10. (a) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.
 - (b) If a Derived Work is distributed under a different license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the Work.
11. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.
12. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless required by applicable law or agreed to in writing will The Copyright Holder, or any au-

thor named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

Maintenance of The Work

The Work has the status ‘author-maintained’ if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that it is ‘author-maintained’.

The Work has the status ‘maintained’ if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status ‘maintained’ to ‘unmaintained’ if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.
2. If this search is successful, then enquire whether the Work is still maintained.
 - (a) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.
 - (b) If the search is unsuccessful or no action to resume active maintenance is taken by the Current

Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a L^AT_EX work, this could be done, for example, by posting to `comp.text.tex`.)

3. (a) If the Current Maintainer is reachable and agrees to pass maintenance of the Work to you, then this takes effect immediately upon announcement.
- (b) If the Current Maintainer is not reachable and the Copyright Holder agrees that maintenance of the Work be passed to you, then this takes effect immediately upon announcement.
4. If you make an ‘intention announcement’ as described in 2b above and after three months your intention is challenged neither by the Current Maintainer nor by the Copyright Holder nor by other people, then you may arrange for the Work to be changed so as to name you as the (new) Current Maintainer.
5. If the previously unreachable Current Maintainer becomes reachable once more within three months of a change completed under the terms of 3b or 4, then that Current Maintainer must become or remain the Current Maintainer upon request provided they then update their communication data within one month.

A change in the Current Maintainer does not, of itself, alter the fact that the Work is distributed under the LPPL license.

If you become the Current Maintainer of the Work, you should immediately provide, within the Work, a prominent and unambiguous statement of your status as Current Maintainer. You should also announce your new status to the same pertinent community as in 2b above.

Whether and How to Distribute Works under This License

This section contains important instructions, examples, and recommendations for authors who are considering distributing their works under this license. These authors are addressed as ‘you’ in this section.

Choosing This License or Another License

If for any part of your work you want or need to use *distribution* conditions that differ significantly from those in this license, then do not refer to this license anywhere in your work but, instead, distribute your work under a different license. You may use the text of this license as a model for your own license, but your license should not refer to the LPPL or otherwise give the impression that your work is distributed under the LPPL.

The document ‘`modguide.tex`’ in the base L^AT_EX distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing L^AT_EX under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated to L^AT_EX, the discussion in ‘`modguide.tex`’ may still be relevant, and authors intending to distribute their works under any license are encouraged to read it.

A Recommendation on Modification Without Distribution

It is wise never to modify a component of the Work, even for your own personal use, without also meeting the above conditions for distributing the modified component. While you might intend that such modifications will never be distributed, often this will happen by accident – you may forget that you have modified that component; or it may not occur to you when allowing others to access the modified version that you are thus distributing it and violating the conditions of this license in ways that could have legal implications and, worse, cause problems for the community. It is therefore usually in your best interest to keep your copy of the Work identical with the public one. Many works provide ways to control the behavior of that work without altering any of its licensed components.

How to Use This License

To use this license, place in each of the components of your work both an explicit copyright notice including your name and the year the work was authored and/or last substantially modified. Include also a statement that the distribution and/or modification of that component is

constrained by the conditions in this license.

Here is an example of such a notice and statement:

```
%% pig.dtx
%% Copyright 2005 M. Y. Name
%
% This work may be distributed and/or modified under the
% conditions of the LaTeX Project Public License, either version 1.3
% of this license or (at your option) any later version.
% The latest version of this license is in
%   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.