

The microtype package

Subliminal refinements towards typographical perfection

R Schlicht
w.m.1@gmx.net

v2.8
2020/12/07

The `microtype` package provides a \LaTeX interface to the micro-typographic extensions that were introduced by `pdfTeX` and have since also propagated to `LuaTeX` and `XYTeX`: 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 `pdfTeX` (version 0.14f or later), `LuaTeX`, or `XYTeX` (at least version 0.9997). Font expansion works with `pdfTeX` (version 1.20 for automatic expansion) or `LuaTeX`. The package will by default enable protrusion and expansion if they can safely be assumed to work. Disabling ligatures requires `pdfTeX` (≥ 1.30) or `LuaTeX`, while the adjustment of interword spacing and of kerning only works with `pdfTeX` (≥ 1.40). Letterspacing is available with `pdfTeX` (≥ 1.40) or `LuaTeX` (≥ 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 © 2004–2020 R Schlicht. It may be distributed and/or modified under the conditions of the [\$\LaTeX\$ Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPPL maintenance status ‘author-maintained’.

Contents

1	Micro-typography with T_EX	4
2	Getting started	5
3	Options	6
3.1	Enabling the micro-typographic features	6
3.2	Character protrusion	7
3.3	Font expansion	7
3.4	Tracking	8
3.5	Miscellaneous options	9
3.6	Changing options later	9
4	Selecting fonts for micro-typography	10
5	Micro fine tuning	12
5.1	Character protrusion	13
5.2	Font expansion	14
5.3	Tracking	15
5.4	Additional kerning	18
5.5	Interword spacing	19
5.6	Character inheritance	20
5.7	Configuration files	20
6	Context-sensitive setup	22
7	Letterspacing revisited	23
8	Disabling ligatures	24
9	Hints and caveats	25
10	Contributions	28
11	Acknowledgments	28
12	References	29
13	Short history	30
14	Implementation	34
14.1	Preliminaries	35
	Debugging [36] Requirements [38] Declarations [42] Auxiliary macros [43] Compatibility [51]	
14.2	Font setup	56
	Protrusion [62] Expansion [69] Interword spacing (glue) [72] Additional kern- ing [73] Tracking [74] Disabling ligatures [85] Loading the configuration [87] Translating characters into slots [91] Hook into L ^A T _E X's font selection [99] Context-sensitive setup [102]	
14.3	Configuration	104
	Font sets [104] Variants and aliases [110] Disabling ligatures [111] Interaction with babel [112] Fine tuning [112] Character inheritance [119] Permuta- tion [121]	

14.4	Package options	124
	Declaring the options [124] Loading the definition file [128] Reading the configuration file [128] Hook for other packages [129] Changing options later [130] Processing the options [132]	
15	Configuration files	142
15.1	Font sets	142
15.2	Font variants and aliases	143
15.3	Interaction with babel	145
15.4	Note on admissible characters	145
15.5	Character inheritance	146
	OT1 [146] T1 [146] LY1 [147] OT4 [148] QX [148] T5 [149] EU1, EU2, TU [149] Euro symbols [150]	
15.6	Tracking	150
15.7	Font expansion	151
15.8	Character protrusion	153
	Normal [153] Italics [162] Small caps [173] Italic small caps [175] Text companion [177] Computer Modern math [182] AMS symbols [185] Euler [189] Euro symbols [193]	
15.9	Interword spacing	194
	Nonfrenchspacing [196]	
15.10	Additional kerning	197
	French [197] Turkish [198]	
16	OpenType configuration files	199
16.1	Character inheritance	199
16.2	Character protrusion	203
17	Auxiliary file for micro fine tuning	216
A	The title logo	218
A.1	Macros	218
A.2	Document	223
B	The letterspacing illustration	223
B.1	Macros	223
B.2	Document	226
C	Change history	228
D	Index	237
E	The L^AT_EX Project Public License	246

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	89

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 X_YT_EX. These features make them the tool of choice not only for the creation of electronic documents but also of works of outstanding time-honoured typography: most prominently, *character protrusion* (also known as margin kerning) and *font expansion*. Quoting Hàn Thế Thành, the author of pdfT_EX, who writes in his thesis:

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

Protrusion *off*
Expansion *off*

Both features are enabled throughout this document.

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

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

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

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

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

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

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

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

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

2 Getting started

There is nothing surprising in loading this package:

```
\usepackage{mimetype}
```

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

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

3 Options

Like many other \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 – if preceded by an asterisk, this default only applies when running an up-to-date `pdfTeX` in PDF mode).

3.1 Enabling the micro-typographic features

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

expansion These are the main options to control the level of micro-typographic refinement which the fonts in your document should gain. By default, the package is moderately greedy: character protrusion will always be enabled, font expansion will only be disabled when the fonts cannot be expanded automatically, that is, with `pdfTeX` versions older than 1.20 or in DVI output mode (see section 3.5), or with `XYTeX`. 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, e.g., 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, ` `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 `XYTeX` (you may use the `LetterSpace` option of the `fontspec` package instead). With `pdfTeX`, it is only available in PDF mode.

kerning `true, false, ` `false`

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

Table 1:

Availability of micro-typographic features

T _E X engine			Micro-typographic features					
Engine	Version	Output	Protrusion	Expansion	(= auto)	Kerning	Spacing	Tracking
pdfT _E X	< 0.14f	DVI/PDF	∅	∅	∅	∅	∅	∅
	≥ 0.14f	DVI/PDF	★	☒	∅	∅	∅	∅
	≥ 1.20	DVI	★	☒	∅	∅	∅	∅
		PDF	★	☒	★	∅	∅	∅
	≥ 1.40	DVI	★	☒	∅	☒	☒	∅
		PDF	★	☒	★	☒	☒	☒
LuaT _E X	≥ 0.30	DVI	★	☒	∅	∅	∅	∅
		PDF	★	☒	★	∅	∅	∅
	≥ 0.62	DVI	★	∅	☒ ^a	∅	∅	☒
		PDF	★	∅	★	∅	∅	☒
X _Y T _E X	≥ 0.9997	PDF	★	∅	∅	∅	∅	

★ = enabled ☒ = not enabled ∅ = not available ^a by means of variable tracking

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

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

3.2 Character protrusion

pdfT_EX 0.14f | LuaT_EX 0.30 | X_YT_EX 0.9997

factor *(integer)* 1000

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

unit character, *(dimension)* character

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

3.3 Font expansion

pdfT_EX 0.14f | LuaT_EX 0.30

auto true, false * true

Beginning with pdfT_EX version 1.20 (inherited by LuaT_EX), the expanded instances of the fonts may be calculated automatically and at run-time instead of the user having to prepare them in advance. This option is true by default provided that you

are using a TeX engine with this capability and the output mode is PDF.² If `auto` is set to `false`, the font instances 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](#)). With LuaTeX, expansion is always automatic.

When generating DVI files, font expansion has to be enabled explicitly. With pdfTeX, *automatic* font expansion will not work because the postprocessing drivers (`dvips`, `dvipdfm`, etc.) resp. the DVI viewer are not able to generate the fonts on the fly. With LuaTeX, on the other hand, expansion in DVI mode is realised by modifying the inter-letter spacing (tracking) instead of the glyphs themselves, which may or may not be desired.

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.³ 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 $+1000$.

² With pdfTeX, automatic font expansion does not work with bitmap fonts. Therefore, if you are using the Computer Modern Roman fonts in T1 encoding, you should either install the `cm-super` package or use the Latin Modern fonts (package `lmodern`).

³ 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`.

3.5 Miscellaneous options

draft true, false false

final If the `draft` option is passed to the package, *all micro-typographic extensions will be disabled*, which may lead to different line, and hence page, breaks. The `draft` and `final` options may also be inherited from the class options; of course, you can override them in the package options. E.g., if you are using the class option `draft` to show any overfull boxes, you should load `microtype` with the `final` option.

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 `verbose` option. When `microtype` encounters a problem that is not fatal (e.g., an unknown character in the settings, or non-existent settings), it will by default only issue a warning and try to continue. Loading the package with `verbose=errors` will turn all warnings into errors, so that you can be sure that no problem will go unnoticed. If on the other hand you have investigated all warnings and decide to ignore them, you may silence `microtype` with `verbose=silent`.

babel true, false false

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

config *(file name)* `microtype`

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

DVIoutput true, false * false

`pdfTeX` and `LuaTeX` are not only able to generate PDF output but can also spit out DVI files. In fact, all recent `TeX` systems are using `pdfTeX` as the default engine also for DVI output, and `LuaTeX` too can be called in DVI mode. However, since changing the output mode inside the document may have undesired effects, this option should be considered deprecated; instead, it is recommended to just call the respective program (`latex` resp. `dviualatex`). For `XYTeX`, this option is not applicable.

3.6 Changing options later

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

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

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

4 Selecting fonts for micro-typography

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

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

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

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

```
\DeclareMicrotypeSet{basictext}
{ encoding = {OT1,T1,T2A,Ly1,OT4,QX,T5,EU1,EU2,TU},
  family   = {rm*,sf*},
  series   = {md*},
  size     = {normalsize,footnotesize,small,large}
}
```

If you now call

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

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

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

```
\DeclareMicrotypeSet{alltext}
{ encoding = {OT1,T1,T2A,Ly1,OT4,QX,T5,TS1,EU1,EU2,TU} }
```

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

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

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

Table 2:

Predefined font sets

Set name	Font attributes				
	Encoding	Family	Series	Shape	Size
all	∅	∅	∅	∅	∅
alltext (allmath)	Text encodings, TS1 (OML, OMS, U)	∅	∅	∅	∅
alltext-nott (allmath-nott)	Text encodings, TS1 (OML, OMS, U)	\rm*, \sf*	∅	∅	∅
basictext (basicmath)	Text encodings (OML, OMS)	\rm*, \sf*	\md*	∅	\normalsize, \footnotesize, \small, \large
smallcaps	Text encodings	∅	∅	\sc*,si,scit	∅
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, TU '\...*' = '\...default'

boundary is included in the range, the upper boundary is not. Thus, '12-16' would match 12 pt, 13.5 pt and 15.999 pt, for example, but not 16 pt. You are allowed to omit the lower or upper bound ('-10', '1arge-').

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

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

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

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

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

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

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

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

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

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

5 Micro fine tuning

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

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

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

The characters may be specified either as a single letter (A), as a text symbol command (`\textquoteleft`), or as a slot number (resp. Unicode number for LuaTeX or XeTeX): three or more digits for decimal notation, prefixed with “ for hexadecimal, with ‘ for octal numerals (e.g., the ‘fl’ ligature in T1 encoding: 029, “1D, ‘35). 8-bit (and even UTF-8) characters may be entered directly or in L^AT_EX’s traditional 7-bit notation: both “A and “A are valid, provided the character is actually declared in both the input and the font encoding. With LuaTeX or XeTeX, you may additionally specify a (font-specific) glyph name, prefixed with ‘/’ (e.g., the ‘fl’ ligature as /f_1). Note that you also have the possibility to declare lists of characters that should inherit settings (see section 5.6).

5.1 Character protrusion

pdfTeX 0.14f | LuaTeX 0.30 | XeTeX 0.9997

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

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

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

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

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

Options:

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

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

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

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

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

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

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

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

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

5.2 Font expansion

pdfTeX 0.14f | LuaTeX 0.30

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

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

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

Options:

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

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

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

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

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

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

This method of employing contexts to temporarily apply different expansion parameters only works with pdfTeX version 1.40.4 or later,⁶ or with LuaTeX. Also note that both pdfTeX and LuaTeX prohibit the use of fonts with different expansion limits or steps (even of different fonts) within one paragraph, hence the `sloppy` context would have 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

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

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

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

Options:

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

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

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

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

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

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

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

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

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

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

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

and then write:

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

this would be the (typographically dubious) outcome:



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

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

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

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

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

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

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

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

5.4 Additional kerning

pdfTeX 1.40

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

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

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

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

Options:

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

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

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

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

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

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

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

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

5.5 Interword spacing

pdfTeX 1.40

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

This command allows you to fine tune the interword spacing (also known as glue). A preliminary remark on what a ‘space’ is may be in order: between two words, 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, but the settings must always contain the two separating commas.

Options:

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

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

```

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

```

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

5.6 Character inheritance

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

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

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

5.7 Configuration files

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

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

`\DeclareMicrotypeVariants` {*list of suffixes*}

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

Table 3:

Fonts with tailored protrusion settings

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

a Incomplete
b Aliases: Latin Modern Roman (lmr), ae (aer), zefonts (zer), eco (cmor), hfoldsty (hfor)
c Aliases: mathdesign/Charter (mdbch), MicroPress's chmath (chr), XCharter
d Settings inherited from italic shape
e Aliases: mathdesign/URW Garamond (mdugm), garamondx (zgm, zgmj)
f Alias: ulgothic (ulg)
g Aliases: pxfonts (pxr), qfonts/QuasiPalatino, T_EX Gyre Pagella (qpl), newpx, FPL Neu (fp9x, fp9j), domitian
h Aliases: txfonts (txr), qfonts/QuasiTimes, T_EX Gyre Termes (qtm), newtx, tempora, step, stix/stix2
i Alias: New Computer Modern
j Aliases: T_EX Gyre Pagella, Palatino LT Std, Palatino, Domitian
k Aliases: Latin Modern (lmsy, lmm)
l Alias: eulervm (zeur, zeus)

same file `mt-pad.cfg`. This command expects a comma-separated list of variant suffixes. The starred version appends the suffix(es) to the existing list. The default declaration in `microtype.cfg` is:

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

`\DeclareMicrotypeAlias` $\{ \langle font\ name \rangle \} \{ \langle alias\ font \rangle \}$

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` $\{\langle font\ name\rangle\}$

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- $\langle font\ name\rangle$.cfg’.

6 Context-sensitive setup

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

`\microtypecontext` $\{\langle context\ assignments\rangle\}$

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

`\begin{microtypecontext}` $\{\langle context\ assignments\rangle\}$

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

`\textmicrotypecontext` $\{\langle context\ assignments\rangle\} \{\langle general\ text\rangle\}$

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{\normal font
  \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{\normal font
  \microtypecontext{protrusion=footnote}\@thefnmark}}}
\let\m@mmf@prepare\relax
\let\m@mmf@check\relax
```

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

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

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

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

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

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

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

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

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

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

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

7 Letterspacing revisited

pdfTeX 1.40 | LuaTeX 0.62

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

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

```
\lsstyle
```

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

```
\textls*
```

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

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

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

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

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

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

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

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

8 Disabling ligatures

pdfTeX 1.30 | LuaTeX 0.30

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

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

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


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

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

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

9 Hints and caveats

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

- ! Font csnameendcsname=*cmr10+20 at 10.0pt* not loadable: Metric (TFM) file not found.
This error message will occur if you are trying to employ font expansion while creating DVI output. Remember that *automatic* font expansion only works when running pdfTeX or LuaTeX in PDF mode. Although expansion is also possible in DVI mode with pdfTeX, 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`.

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

- Warning: pdf_latex (file *ecrm1000+20*): Font *ecrm1000+20 at 1200* not found
Furthermore, pdf_TE_X 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 pdf_TE_X versions, this is only possible if you manually create expanded instances of the fonts.
- ! Font *T1/cmr/m/n/10=ecrm1000 at 10.0pt* not loaded: Not enough room left.
Memory parameter ‘font_{_}mem_{_}size’ too small.
- ! TeX capacity exceeded, sorry [maximum internal font number (font_{_}max)=2000].
Memory parameter ‘font_{_}max’ too small.
- ! TeX capacity exceeded, sorry [PDF memory size (pdf_{_}mem_{_}size)=65536].
Memory parameter ‘pdf_{_}mem_{_}size’ too small (pdf_TE_X versions older than 1.30).
When applying micro-typographic enhancement to a large document with a lot of fonts, pdf_TE_X 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., T_EX Live, change the settings in *texmf.cnf*, for MiK_TE_X, in the file *miktex.ini* (2.4 or older) resp. *pdf_latex.ini* (2.5 or newer).
- pdf_Te_X warning (font expansion): font should be expanded before its first use
This warning will occur with pdf_TE_X versions older than 1.40.4, if tracking *and* expansion is applied to a font. It is harmless and can be ignored.

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

10 Contributions

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

11 Acknowledgments

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

Harald Harders has contributed protrusion settings for Adobe Minion. I would also like to thank him for a number of bug reports and suggestions he had to make. *Andreas Böhmann* has suggested the possibility to specify ranges of font sizes, and resourcefully assisted in implementing this. He also came up with some good ideas for the management of complex configurations. *Ulrich Dirr* has made numerous suggestion, especially concerning the new extensions of interword spacing adjustment and additional character kerning. *Georg Duffner* has patiently tested *microtype* under $\text{X}_{\text{T}}\text{E}_{\text{X}}$ and $\text{L}_{\text{u}}\text{a}\text{T}_{\text{E}}\text{X}$ with his beautiful OpenType font EB Garamond¹³. My thanks also go to *Maciej Eder* for contributing settings for the QX encoding, as well as to *Karl Karlsson* for providing settings for the Cyrillic T2A encoding, and to *Hendrik Vogt*, who made substantial improvements to the Computer Modern Roman italic settings. I thank *Loren B. Davis* for providing protrusion settings for OpenType versions of Palatino Linotype. I am also very much indebted to *Élie Roux*, who not only contributed the `lua` module in the first place, but also, together with *Philipp Gesang*, took care of updating it for the developments in $\text{L}_{\text{u}}\text{a}\text{T}_{\text{E}}\text{X}$ land.

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

Additionally, the following people have reported bugs, made suggestions or helped otherwise (in chronological order, quotes indicate **TeX.SX** user names): *Tom Kink*, *Herb Schulz*, *Michael Hoppe*, *Gary L. Gray*, *Georg Verweyen*, *Christoph Bier*, *Peter Muthesius*, *Bernard Gaille*, *Adam Kucharczyk*, *Mark Rossi*, *Stephan Hennig*, *Michael Zedler*, *Herbert Voß*, *Ralf Stubner*, *Holger Uhr*, *Peter Dyballa*, *Morten Høgholm*, *Steven Bath*, *Daniel Flipo*, *Michalis Miatidis*, *Sven Naumann*, *Ross Hetherington*, *Wiebke Petersen*, *Geoff Vallis*, *Steven E. Harris*, *Karl Berry*, *Peter Meier*, *Nathan Rosenblum*, *Wolfram Schaalo*, *Vasile Gaburici*, *Sveinung Heggen*, *Axel Berger*, *Colin Rourke*, *Maverick Woo*, *Silas S. Brown*, *Lars Rönnbäck*, *Christian Stark*, *Leo*, *Marcin Borkowski*, *hscm*, *George Gratzner*, *Josep Maria Font*, *Juan Acevedo*, *Heiko Oberdiek*, *Till A. Heilmann*, *Rolf Dieterich*, *Seamus Bradley*, *Meho R*, *Steffen Hoffmann*, *Scott Pakin*, *Maïeul Rouquette*, *Jonas Hogstrom*, *Gabriel Kerneis*, ‘*RazorXsr*’, *Sebastian Schubert*, ‘*Dave*’, *Giuseppe Palma*, *Stephan Stiller*, *Christopher Schramm*, ‘*uli*’, *Sam Mason*, ‘*kleenstar*’, ‘*Henning*’, *Ronnie Marks*, *David Carlisle*, ‘*Max*’, ‘*HcN*’, *Will Robertson*, ‘*user11126*’, *Ulrike Fischer*, ‘*Daniel*’, ‘*ltcomdata*’, *Reinhard Kotucha*, ‘*jcr*’, *Nils Anders Danielsson*, *Paolo Ney*, *Frank Mittelbach*, *Franz Wexler*, *Moritz Wemheuer*, ‘*Andy N*’, *Phelype Oleinik*, *Falk Hanisch*, *Markus Kohm*, *Paolo Polesana*, *Oliver Kopp*, *Hironori Kitagawa*, ‘*d909*’, *Daniel Benjamin Miller*, *Md Ayquassar* and *Marcel Krüger*.

12 References

Hàn Thế Thành, ‘Micro-typographic extensions to the $\text{T}_{\text{E}}\text{X}$ typesetting system’, Diss. Masaryk University Brno 2000, in: *TUGboat*, vol. 21 (2000), no. 4, pp. 317–434. (Online at <http://www.tug.org/TUGboat/Articles/tb21-4/tb69thanh.pdf>)

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

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

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

Hàn Thế Thành, Sebastian Rahtz, Hans Hagen, Hartmut Henkel, Paweł Jackowski, Martin Schröder, Karl Berry, *The pdfTeX user manual*, 2 March 2020. (Available from CTAN at </systems/doc/pdftex/manual/pdftex-a.pdf>)

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

L^AT_EX3 Project Team, *L^AT_EX 2_ε font selection*, July 2020. (Available from CTAN at </macros/latex/base/fntguide.pdf>)

Will Robertson, *The fontspec package: Font selection for X_YL^AT_EX and LuaL^AT_EX*, 21 February 2020. (Available from CTAN at <pkg/fontspec>)

Élie Roux, Khaled Hosny, Philipp Gesang, Ulrike Fischer, Marcel Krüger, *The luaotfload package*, 2 September 2020. (Available from CTAN at <pkg/luaotfload>)

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

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

13 Short history

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

2.8 (2020/12/07)

- New default font sets for expansion and spacing: ‘alltext-nott’ [4, table 2]

2.7 (2017/07/07)

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

2.6 (2016/05/01)

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

2.5 (2013/03/13)

- Support for the `fontspec` package, viz. for OpenType fonts with LuaTeX and X_YL^AT_EX
- Support for protrusion with X_YL^AT_EX ≥ 0.9997

- Support for tracking/letterspacing with LuaTeX ≥ 0.62
- Allow context-sensitive setup with LuaTeX
- Info if protrusion settings are generic
- Protrusion settings for Latin Modern Roman (OpenType)
- Protrusion settings for Charis SIL (OpenType)
- Protrusion settings for Palatino Linotype (OpenType)

2.4 (2010/01/10)

- Protrusion settings for T2A encoded Minion

2.3e (2009/11/09)

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

2.3d (2009/03/27)

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

2.3c (2008/11/11)

- Support for LuaTeX enabled by default

2.3 (2007/12/23)

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

2.2 (2007/07/14)

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

2.1 (2007/01/21)

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

2.0 (2007/01/14)

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

- New package ‘letterspace’ providing the commands for robust and hyphenatable letterspacing [7]

1.9e (2006/07/28)

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

1.9d (2006/05/05)

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

1.9c (2006/02/02)

- Protrusion settings for URW Garamond

1.9a (2005/12/05)

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

1.9 (2005/10/28)

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

1.8 (2005/06/23)

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

1.7 (2005/03/23)

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

1.6 (2005/01/24)

- New option ‘factor’ to influence protrusion resp. expansion of all characters of a font or font set [3.2, 5]

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

These are all commands for the outside world. We define them here as blank
commands, so that they won't generate an error if we are not running pdfTeX.
31 (package)
32 \newcommand*\DeclareMicrotypeSet[3] [] {}
33 \newcommand*\UseMicrotypeSet[2] [] {}
34 \newcommand*\DeclareMicrotypeSetDefault[2] [] {}
35 \newcommand*\SetProtrusion[3] [] {}
36 \newcommand*\SetExpansion[3] [] {}
37 \newcommand*\SetTracking[3] [] {}
38 \newcommand*\SetExtraKerning[3] [] {}
39 \newcommand*\SetExtraSpacing[3] [] {}
40 \newcommand*\DisableLigatures[2] [] {}
41 \newcommand*\DeclareCharacterInheritance[3] [] {}
42 \newcommand*\DeclareMicrotypeVariants[1] {}
43 \newcommand*\DeclareMicrotypeAlias[2] {}
44 \newcommand*\LoadMicrotypeFile[1] {}
45 \newcommand*\DeclareMicrotypeBabelHook[2] {}
46 \newcommand*\microtypesetup[1] {}
47 \newcommand*\microtypecontext[1] {}
48 \newcommand*\textmicrotypecontext[2] {#2}
49 \ifpackageloaded{letterspace}{\let\MT@textls\relax}{%
50 (package)
51 \newcommand*\lsstyle{}
52 \newcommand\textls[2] [] {}
53 \def\textls#1#{}

```

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

These commands also have a starred version.

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

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

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

Don't load letterspace.

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

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

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

`\MT@warning` Communicate.

```
\MT@warning@nl 76 \def\MT@warning{\PackageWarning\MT@MT}
\MT@info 77 \def\MT@warning@n1#1{\MT@warning{#1\@gobble}}
78 < *package >
\MT@info@nl 79 \def\MT@info{\PackageInfo\MT@MT}
\MT@vinfo 80 \def\MT@info@n1#1{\MT@info{#1\@gobble}}
\MT@error 81 \let\MT@vinfo\@gobble
\MT@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@info 0: almost none
\MT@info@n1 1: + sets & lists
2: + heirs
3: + slots
4: + factors
```

```
87 < *debug >
88 \MT@warning@n1{This is the debug version}
89 \newcount\tracingmicrotype
```

```

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

```

\tracingmicrotypeinpdf

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

Cases for \tracingmicrotypeinpdf:

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

```
98 \newcount\tracingmicrotypeinpdf
```

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

```
\tracingmicrotypeinpdf=2
```

```

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

```

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

```

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

```

\iftracingmicrotypeinpdfall

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

```
105 \newif\iftracingmicrotypeinpdfall
```

\MT@show@pdfannot

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

```

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

```

14.1.2 Requirements

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

- 0: miniltx
- 1: eplain
- 2: 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\epain@undefined
133     \def\MT@plain{0}
134     \def\PackageWarning#1#2{%
135       \begingroup
136         \newlinechar=10 %
137         \def\MessageBreak{^^J(#1)\@spaces\@spaces\@spaces\@spaces}%
138         \immediate\write16{^^JPackage #1 Warning: #2\on@line.^^J}%
139       \endgroup
140     }
141     \def\on@line{ on input line \the\inputlineno}
142     \def\@spaces{\space\space\space\space}
143   \fi
144 \fi

```

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

```

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

```

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 <letterspace>^^Q\MT@warning@n1{This package requires the etex extensions.
159 <letterspace>^^Q \MessageBreak Exiting}\MT@restore@catcodes\endinput
160 <debug>\MT@dinfor@n1{0}{this is
161 <debug>^^Q not
162 <debug> etex}

```

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

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

```

163 \def\MT@clear@options{%
164 <plain> \MT@requires@latex1{%
165   \AtEndOfPackage{\let\unprocessedoptions\relax\MT@restore@catcodes}%
166   \let\CurrentOption@empty
167 <package> \let\MT@endinput\endinput
168 <plain> }\relax
169 }

```

A hack circumventing the T_EX Live 2004 hack which undefines the pdfT_EX primitives in the format in order to hide the fact that pdfT_EX is being run from the user. This has been *fixed* in T_EX Live 2005.

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

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

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

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

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

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

Currently, we have to distinguish seven cases for pdfT_EX:

- 0: not running pdfT_EX
- 1: pdfT_EX (< 0.14f)
- 2: + micro-typographic extensions (0.14f,g)
- 3: + protrusion relative to 1 em (≥ 0.14h)

- 4: + automatic font expansion; protrusion no longer has to be set up first; scale factor fixed to 1000; default `\efcode = 1000` (≥ 1.20)
- 5: + `\(left,right)marginkern`; `\pdfnoligatures`; `\pdfstrcmp`; `\pdfescapestring` (≥ 1.30)
- 6: + adjustment of interword spacing; extra kerning; `\letterspacefont`; `\pdfmatch`¹⁴; `\pdftracingfonts`; always e-TeX (≥ 1.40)
- 7: + `\letterspacefont` doesn't disable ligatures and kerns; `\pdfcopyfont` ($\geq 1.40.4$)

```

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

```

`\MT@xetexno` X_YTeX supports character protrusion since version 0.9997.

```

244 (*xetex-def)
245 (debug)\MT@dinfol{0}{this is xetex (\the\XeTeXversion\XeTeXrevision)}
246 \ifdim 0\XeTeXrevision pt < 0.9997pt
247   \def\MT@xetexno{1}
248 \else
249   \def\MT@xetexno{2}
250 \fi
251 (debug)\MT@dinfol{0}{xetex no.: \MT@xetexno}
252 (/xetex-def)

```

`\MT@luatexno` Cases for LuaTeX (`\luatexversion` ought to have been enabled by the format):

- 0: N/A
- 1: LuaTeX (< 0.36)
- 2: + `\directlua` without state number (≥ 0.36)

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

3: + `\letterspacefont`; non-automatic expansion doesn't work anymore, and automatic expansion in DVI mode is realised by modifying the tracking, not the glyphs¹⁵ (≥ 0.62)

4: + almost all of the pdfTeX primitives have been renamed (≥ 0.85)

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

Also, sometime between 1.0.4 and 1.0.7, the function `font.setexpansion` has been introduced, but we'll test this directly later.

```

253 (*luatex-def)
254 (debug)\MT@dinfolo{this is luatex (\the\luatexversion)}
\MT@lua    Communicate with lua. Beginning with LuaTeX 0.36, \directlua no longer requires
           a state number.
255 \let\MT@lua\directlua
256 \def\MT@luatex@no{5}
257 \ifnum\luatexversion<90
258   \def\MT@luatex@no{4}
259   \ifnum\luatexversion<85
260     \def\MT@luatex@no{3}
261     \ifnum\luatexversion<62
262       \def\MT@luatex@no{2}
263       \ifnum\luatexversion<36
264         \def\MT@lua{\directlua0}
265         \def\MT@luatex@no{1}
266       \fi
267     \fi
268   \fi
269 \fi
270 (debug)\MT@dinfolo{0}{luatex no.: \MT@luatex@no}
271 (/luatex-def)
272 (*pdftex-def|xetex-def|letterspace)
273 \ifnum
274 (pdftex-def|xetex-def) \csname MT@MT@engine tex@no\endcsname < 2
275 (letterspace) \MT@engine@toold=\z@
276 \MT@warning@nl{You
277 (*letterspace)
278   \ifx\MT@engine\relax
279     don't seem to be using pdftex or luatex.\MessageBreak
280     Try running `pdftex' or `luatex' instead of.\MessageBreak
281     \ifx\XeTeXversion\undefined\else xe\fi tex'%
282   \else
283 (/letterspace)
284     are using a \MT@engine tex version older than
285 (pdftex-def)      0.14f%
286 (xetex-def)      0.9997%
287 (letterspace)    \MT@pdf@or@lua{1.40}{0.62}%
288     .\MessageBreak
289     ~\MT@MT' does not work with this version.\MessageBreak
290     Please install a newer version of \MT@engine tex%
291 (letterspace)    \fi
292     .\MessageBreak I will quit now}
293 \MT@clear@options
294 \endinput\fi
295 (/pdftex-def|xetex-def|letterspace)

```

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

```

296 (*package|letterspace)

```

15 This may have been changed earlier, but I'm no longer able to find out when (the last version that actually works for me is 0.40).

```

297 \RequirePackage{keyval}[1997/11/10]
298 <package>
\MT@toks    We need a token register.
299 \newtoks\MT@toks
\ifMT@if@   A scratch if.
300 \newif\ifMT@if@

```

14.1.3 Declarations

```

\ifMT@protrusion    These are the global switches ...
\ifMT@expansion     301 \newif\ifMT@protrusion
\ifMT@auto          302 \newif\ifMT@expansion
\ifMT@selected      303 \newif\ifMT@auto
\ifMT@selected      304 \newif\ifMT@selected
\ifMT@noligatures   305 \newif\ifMT@noligatures
\ifMT@draft         306 \newif\ifMT@draft
\ifMT@spacing       307 \newif\ifMT@spacing
\ifMT@kerning       308 \newif\ifMT@kerning
\ifMT@tracking      309 \newif\ifMT@tracking
\ifMT@tracking      310 \newif\ifMT@babel
\ifMT@babel         [This line intentionally left blank.]
\MT@pr@level        ... and numbers.
\MT@ex@level        311 \let\MT@pr@level\tw@
\MT@pr@factor        312 \let\MT@ex@level\tw@
\MT@ex@factor        313 \let\MT@pr@factor\@m
\MT@sp@factor        314 \let\MT@ex@factor\@m
\MT@kn@factor        315 \let\MT@sp@factor\@m
\MT@kn@factor        316 \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         317 \let\MT@pr@unit\@empty
                    318 \let\MT@sp@unit\@one
                    319 \def\MT@kn@unit{1em}
\MT@stretch         Expansion settings.
\MT@shrink          320 \let\MT@stretch\@m@ne
\MT@step            321 \let\MT@shrink \@m@ne
                    322 \let\MT@step \@m@ne
\MT@pr@min          Minimum and maximum values allowed by pdfTEX.
\MT@pr@max          323 \def\MT@pr@min{-\@m}
\MT@ex@min          324 \let\MT@pr@max\@m
                    325 \let\MT@ex@min\z@
\MT@ex@max          326 \let\MT@ex@max\@m
\MT@sp@min          327 \def\MT@sp@min{-\@m}
\MT@sp@max          328 \let\MT@sp@max\@m
\MT@kn@min          329 \def\MT@kn@min{-\@m}
\MT@kn@max          330 \let\MT@kn@max\@m
\MT@kn@max          331 </package>
\MT@tr@min          332 \def\MT@tr@min{-\@m}
\MT@tr@max          333 \let\MT@tr@max\@m
                    334 <package>
\MT@factor@default Default factor.
                    335 \def\MT@factor@default{1000 }
\MT@stretch@default Default values for expansion.
\MT@shrink@default 336 \def\MT@stretch@default{20 }

```

```

337 \def\MT@shrink@default{20 }
\MT@letterspace      Default value for letterspacing (in thousandths of 1 em).
\MT@letterspace@default 338 </package>
339 \let\MT@letterspace\m@ne
340 \def\MT@letterspace@default{100}
341 <*package>
\ifMT@document      Our private test whether we're still in the preamble.
342 \newif\ifMT@document
343 </package>
344 </package|letterspace>

```

14.1.4 Auxiliary macros

```

\MT@requires@pdftex  For definitions that depend on a particular pdfTeX resp. LuaTeX version.
\MT@requires@luatex 345 <*pdftex-def|luatex-def>
346 \def
347 <pdftex-def> \MT@requires@pdftex%
348 <luatex-def> \MT@requires@luatex%
349 #1{\ifnum
350 <pdftex-def> \MT@pdftex@no
351 <luatex-def> \MT@luatex@no
352 <#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi>
353 <luatex-def&debug>\MT@requires@luatex4{\MT@lua{tex.enableprimitives('pdf',{'tracingfonts'})}}\relax
354 <pdftex-def&debug>\MT@requires@pdftex6{
355 <debug>\pdftracingfonts=1
356 <pdftex-def&debug>}\relax
357 </pdftex-def|luatex-def>

```

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

```

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

```

We load `luaotfload`, because some of its functions are required in `microtype.lua`. This eliminates the need for the user to load `fontspec` before `microtype`. There will hardly be any LuaTeX documents that don't load this package, anyway. Since 2017/01/01, it is already loaded in the format.

```

360 \ifl@t@r\fmtversion{2017/01/01}\relax{\RequirePackage{luaotfload}}
361 \MT@lua{require("microtype")}
362 </luatex-def>

```

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

```

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

```

```

380 tex.sprint(catpackage, ...)
381 end
382
383 </luafile>

```

To be continued, but first back to primitives.

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

```

384 <*package|letterspace>
385 \def\MT@gllet{\global\let}

```

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

```

386 \def\MT@exp@cs#1#2{\expandafter#1\csname#2\endcsname}
387 <*package>
388 \def\MT@exp@gcs#1#2{\begingroup\expandafter\endgroup\expandafter#1\csname#2\endcsname}

```

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

```

\MT@gdef@n 389 \def\MT@def@n{\MT@exp@cs\def}
390 \def\MT@gdef@n{\MT@exp@gcs\gdef}

```

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

```

\MT@xdef@n 391 </package>
392 \def\MT@edef@n{\MT@exp@cs\edef}
393 <*package>
394 \def\MT@xdef@n{\MT@exp@gcs\xdef}

```

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

```

\MT@gllet@nc 395 \def\MT@let@nc{\MT@exp@cs\let}
396 \def\MT@gllet@nc{\MT@exp@gcs\MT@gllet}

```

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

```

397 </package>
398 \def\MT@let@cn#1#2{\expandafter\let\expandafter#1\csname #2\endcsname}
399 <*package>

```

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

```

\MT@gllet@nn 400 \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
401 \def\MT@gllet@nn{\MT@exp@gcs{\global\expandafter\MT@let@cn}}

```

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

```

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

```

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

```

403 </package>
404 \def\MT@exp@one@n#1#2{\expandafter#1\expandafter{#2}}

```

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

```

405 \def\MT@exp@two@c#1{\expandafter\expandafter\expandafter#1\expandafter}
406 <*package>

```

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

```

407 \def\MT@exp@two@n#1#2#3{%
408   \expandafter\expandafter\expandafter
409   #1\expandafter\expandafter\expandafter
410   {\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 411 \def\MT@ifdefined@c@T#1{%
412   ^^X \ifdefined#1\expandafter\@firstofone\else\expandafter\@gobble\fi

```

```

413 ^^Q \ifx#1\@undefined\expandafter@gobble\else\expandafter\@firstofone\fi
414 }
415 </package>
416 \def\MT@ifdefined@c@TF#1{%
417 ^^X \ifdefined#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
418 <package>^^Q \ifx#1\@undefined
419 <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
420 }
421 \def\MT@ifdefined@n@T#1{%
422 ^^X \ifcsname#1\endcsname\expandafter\@firstofone\else\expandafter\@gobble\fi
423 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
424 <package>^^Q \expandafter\@gobble\else\expandafter\@firstofone\fi
425 }
426 \def\MT@ifdefined@n@TF#1{%
427 ^^X \ifcsname#1\endcsname\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
428 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
429 <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
430 }
431 <*package>

```

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

`\MT@detokenize@c`
`\MT@rem@last@space`

```

432 \def\MT@detokenize@n#1{%
433 ^^X \expandafter\MT@rem@last@space\detokenize{#1} \@ni1
434 ^^Q \string#1%
435 }
436 \def\MT@detokenize@c#1{%
437 ^^X \MT@exp@one@n\MT@detokenize@n#1%
438 ^^Q \MT@exp@two@c\MT@rem@last@space\strip@prefix\meaning#1 \@ni1
439 }
440 \def\MT@rem@last@space#1 #2{#1%
441 \ifx\@nil#2\else \space
442 \expandafter\MT@rem@last@space\expandafter#2\fi
443 }

```

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

```

444 </package>
445 \begingroup
446 \catcode`\%=12
447 \catcode`\&=14
448 \gdef\MT@ifempty#1{&
449 \if %#1%&
450 \expandafter\@firstoftwo
451 \else
452 \expandafter\@secondoftwo
453 \fi
454 }
455 \endgroup
456 <*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).

```

457 </package>
458 </package|letterspace>
459 <pdftex-def>\MT@requires@pdftex6{
460 <letterspace>\MT@pdf@or@lua{
461 <*pdftex-def|letterspace>
462 \def\MT@ifint#1{%
463 \ifcase\pdfmatch{^-*[0-9]+ *$}{#1}\relax
464 \expandafter\@secondoftwo
465 \else
466 \expandafter\@firstoftwo

```

```

467 \fi
468 }
469 }{
470 /pdftex-def|letterspace)
471 (*pdftex-def|xetex-def|letterspace)
472 \def\MT@ifint#1{%
473   \if!\ifnum9<1#1!\else?\fi
474   \expandafter\@firstoftwo
475   \else
476   \expandafter\@secondoftwo
477   \fi
478 }
479 /pdftex-def|xetex-def|letterspace)
480 (pdftex-def|letterspace)}
481 (luatex-def)\def\MT@ifint#1{\csname\MT@lua{microtype.if_int([[#1]])}\endcsname}
482 (*luafile)
483 local function if_int(s)
484   if find(s,"^-[0-9]+ *$") then
485     tex_write("@firstoftwo")
486   else
487     tex_write("@secondoftwo")
488   end
489 end
490 microtype.if_int = if_int
491
492 (luafile)

```

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

```

493 (*pdftex-def)
494 \MT@requires@pdftex6{
495 \def\MT@ifdimen#1{%
496   \ifcase\pdfmatch{^[0-9]+([.][0-9]+)?|[.][0-9]+}%
497     (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
498   \expandafter\@secondoftwo
499   \else
500   \expandafter\@firstoftwo
501   \fi
502 }
503 }{
504 /pdftex-def)
505 (*pdftex-def|xetex-def)
506 \def\MT@ifdimen#1{%
507   \setbox\z@=\hbox{%
508     \MT@count=1#1\relax
509     \ifnum\MT@count=\@ne
510       \aftergroup\@secondoftwo
511     \else
512       \aftergroup\@firstoftwo
513     \fi
514   }%
515 }
516 /pdftex-def|xetex-def)
517 (pdftex-def)
518 (luatex-def)\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen([[#1]])}\endcsname}
519 (*luafile)
520 local function if_dimen(s)
521   if (find(s, "^-[0-9]+(%a*) *$") or
522     find(s, "^-[0-9]*[.][0-9]+(%a*) *$")) then
523     tex_write("@firstoftwo")
524   else
525     tex_write("@secondoftwo")
526   end
527 end
528 microtype.if_dimen = if_dimen

```

```

529
530 </luafile>
\MT@ifdim    Compare floating point numbers.
531 <*package>
532 \def\MT@ifdim#1#2#3{%
533   \ifdim #1\p@ #2 #3\p@
534     \expandafter\@firstoftwo
535   \else
536     \expandafter\@secondoftwo
537   \fi
538 }
539 </package>
\MT@ifstreq  Test whether two strings (fully expanded) are equal.
540 <pdfTex-def><xetex-def>
541 <pdfTex-def>\MT@requires@pdfTex5{
542 \def\MT@ifstreq#1#2{%
543 <pdfTex-def>   \ifnum\pdfstrcmp{#1}{#2}=\z@
544 <xetex-def>   \ifnum\strcmp{#1}{#2}=\z@
545   \expandafter\@firstoftwo
546   \else
547     \expandafter\@secondoftwo
548   \fi
549 }
550 </pdfTex-def><xetex-def>
551 <*pdfTex-def>
552 }{
553 \def\MT@ifstreq#1#2{%
554   \edef\MT@res@a{#1}%
555   \edef\MT@res@b{#2}%
556   \ifx\MT@res@a\MT@res@b
557     \expandafter\@firstoftwo
558   \else
559     \expandafter\@secondoftwo
560   \fi
561 }
562 }
563 </pdfTex-def>
564 <luaTex-def>\def\MT@ifstreq#1#2{\csname\MT@lua{microtype.if_str_eq([[#1]], [[#2]])}\endcsname}
565 <*luafile>
566 local function if_str_eq(s1, s2)
567   if s1 == s2 then
568     tex_write("@firstoftwo")
569   else
570     tex_write("@secondoftwo")
571   end
572 end
573 microtype.if_str_eq = if_str_eq
574
575 </luafile>
\MT@xadd    Add item to a list.
576 <*package>
577 \def\MT@xadd#1#2{%
578   \ifx#1\relax
579     \xdef#1{#2}%
580   \else
581     \xdef#1{#1#2}%
582   \fi
583 }
\MT@xaddb   Add item to the beginning.
584 \def\MT@xaddb#1#2{%
585   \ifx#1\relax

```

```

586     \xdef#1{#2}%
587     \else
588     \xdef#1{#2#1}%
589     \fi
590 }
591 </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@    592 <*package|letterspace>
\MT@clist@function 593 \def\MT@map@clist@n#1#2{%
\MT@clist@break    594   \ifx\@empty#1\else
                    595   \def\MT@clist@function##1{#2}%
                    596   \MT@map@clist@#1,\@nil,\@nnil
                    597   \fi
                    598 }

                    599 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}
                    600 \def\MT@map@clist@#1,{%
                    601   \ifx\@nil#1%
                    602     \expandafter\MT@clist@break
                    603   \fi
                    604   \MT@clist@function{#1}%
                    605   \MT@map@clist@
                    606 }
                    607 \let\MT@clist@function\@gobble
                    608 \def\MT@clist@break#1\@nnil{}
                    609 <*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@    610 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break    611 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
                    612 \def\MT@map@tlist@#1#2{%
                    613   \ifx\@nnil#2\else
                    614     #1{#2}%
                    615     \expandafter\MT@map@tlist@
                    616     \expandafter#1%
                    617   \fi
                    618 }
                    619 \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@    620 \newif\ifMT@inlist@
                    621 \def\MT@in@clist@#1#2{%
                    622   \def\MT@res@a##1,#1,##2##3\@nnil{%
                    623     \ifx##2\@empty
                    624       \MT@inlist@false
                    625     \else
                    626       \MT@inlist@true
                    627     \fi
                    628   }%
                    629   \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil
                    630 }

\MT@rem@from@clist    Remove item <#1> from comma list <#2>. This is basically \@removeelement from
                    ltcntrl.dtx. Using \pdfmatch and \pdflastmatch here would be really slow!
                    631 \def\MT@rem@from@clist#1#2{%
                    632   \def\MT@res@a##1,#1,##2\MT@res@a{##1,##2\MT@res@b}%
                    633   \def\MT@res@b##1,\MT@res@b##2\MT@res@b{\ifx,##1\@empty\else##1\fi}%
                    634   \xdef#2{\MT@exp@two@c\MT@res@b\MT@res@a\expandafter,#2,\MT@res@b,#1,\MT@res@a}%
                    635 }

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

```


than this solution.

```

636 \def\MT@in@tlist#1#2{%
637   \MT@inlist@false
638   \def\MT@res@a{#1}%
639   \MT@map@tlist@c#2\MT@in@tlist@
640 }
641 \def\MT@in@tlist@#1{%
642   \edef\MT@res@b{#1}%
643   \ifx\MT@res@a\MT@res@b
644     \MT@inlist@true
645     \expandafter\MT@tlist@break
646   \fi
647 }

```

`\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@@ 648 \def\MT@in@rlist#1{%
\MT@size@name 649   \MT@inlist@false
650   \MT@map@tlist@c#1\MT@in@rlist@
651 }
652 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
653 \def\MT@in@rlist@@#1#2#3{%
654   \MT@ifdim{#2}=\m@ne{%
655     \MT@ifdim{#1}=\MT@size
656     \MT@inlist@true
657     \relax
658   }%
659   \MT@ifdim\MT@size<{#1}\relax{%
660     \MT@ifdim\MT@size<{#2}%
661     \MT@inlist@true
662     \relax
663   }%
664 }%
665 \ifMT@inlist@
666   \def\MT@size@name{#3}%
667   \expandafter\MT@tlist@break
668 \fi
669 }

```

`\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@iterate
\MT@repeat 670 </package>
671 \def\MT@loop#1\MT@repeat{%
672   \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
673   \MT@iterate \let\MT@iterate\relax
674 }
675 \let\MT@repeat\fi

```

`\MT@while@num` Execute `<#3>` from `<#1>` up to (excluding) `<#2>` (much faster than L^AT_EX's `\@whilenum`).

```

676 \def\MT@while@num#1#2#3{%
677   \@tempcnta#1\relax
678   \MT@loop #3%
679   \advance\@tempcnta \@ne
680   \ifnum\@tempcnta < #2\MT@repeat
681 }
682 </package|letterspace>

```

`\MT@if@luaotf@font` For fonts loaded by `luaotfload` we query the font's table.

```

683 <letterspace>\MT@pdf@or@lua{\let\MT@if@luaotf@font\@secondoftwo}{
684 <luatex-def|letterspace>\def\MT@if@luaotf@font{\csname\MT@lua{%
685 <luatex-def> microtype.if_luaotf_font()
686 <*luafile|letterspace>
687 <luafile>local function if_luaotf_font()
688   local thefont = font.getfont(font.current())

```

```

689 if thefont and ( thefont.format == "opentype" or thefont.format == "truetype" )
690   then tex.write("@firstoftwo")
691   else tex.write("@secondoftwo")
692   end
693 <luafile>end
694 <luafile>microtype.if_luaotf_font = if_luaotf_font
695 <luafile>
696 <luafile|letterspace>
697 <luatex-def|letterspace> }\endcsname
698 <luatex-def|letterspace>
699 <letterspace>

```

`\MT@do@font` Execute $\langle \#1 \rangle$ 256 times,

```
700 <pdfTeX-def|letterspace> \def\MT@do@font{\MT@while@num\z@\cc@lvi}
```

resp. for the whole font for LuaTeX, if it's a Unicode font.

```

701 <*luatex-def>
702 \def\MT@do@font#1{%
703   \MT@if@luaotf@font{%
704     \def\MT@dofont@function#1{%
705       \MT@lua{microtype.do_font()}%
706     }{\MT@while@num\z@\cc@lvi#1}%
707   }
708 </luatex-def>

```

This is the lua function, which is much faster than looping through all glyphs in TeX. Legacy fonts (which this function should never work on) don't contain a `v.index` field.

```

709 <*luafile>
710 local function do_font()
711   local thefont = font.getfont(font.current())
712   if thefont then
713     for i,v in next,thefont.characters do
714       if v.index == nil or v.index > 0 then
715         microtype.sprint([[ \@tempcnta=]] .. i .. [[\relax\MT@dofont@function]])
716       end
717     end
718   end
719 end
720 microtype.do_font = do_font
721
722 </luafile>

```

The XeTeX variant (it's slow ...!).

```

723 <*xetex-def>
724 \def\MT@do@font#1{%
725   \@tempcnta=\z@
726   \MT@loop
727     \iffontchar\MT@font\@tempcnta #1\fi
728     \advance\@tempcnta\@ne
729     \ifnum\@tempcnta < \XeTeXlastfontchar\MT@font \MT@repeat
730   }
731 </xetex-def>
732 <*package>

```

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

`\MT@increment`

```

733 \newcount\MT@count
734 \def\MT@increment#1{%
735   ^^X \edef#1{\number\numexpr #1 + 1\relax}%
736   ^^Q \MT@count=#1\relax
737   ^^Q \advance\MT@count \@ne
738   ^^Q \edef#1{\number\MT@count}%
739 }

```

`\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.

```

740 \def\MT@scale#1#2#3{%
741 ^^Q \multiply #1 #2\relax
742 \ifnum #3 = \z@
743 ^^X #1=\numexpr #1 * #2\relax
744 \else
745 ^^X #1=\numexpr #1 * #2 / #3\relax
746 ^^Q \divide #1 #3\relax
747 \fi
748 }

```

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

```

\MT@abbr@pr@ex
\MT@abbr@pr@c 749 \def\MT@abbr@pr{protrusion}
\MT@abbr@ex@c 750 \def\MT@abbr@ex{expansion}
751 \def\MT@abbr@pr@c{protrusion codes}
\MT@abbr@pr@inh 752 \def\MT@abbr@ex@c{expansion codes}
\MT@abbr@ex@inh 753 \def\MT@abbr@pr@inh{protrusion inheritance}
\MT@abbr@n1 754 \def\MT@abbr@ex@inh{expansion inheritance}
755 \def\MT@abbr@n1{no ligatures}
\MT@abbr@sp 756 \def\MT@abbr@sp{spacing}
\MT@abbr@sp@c 757 \def\MT@abbr@sp@c{interword spacing codes}
758 \def\MT@abbr@sp@inh{interword spacing inheritance}
\MT@abbr@sp@inh 759 \def\MT@abbr@kn{kerning}
\MT@abbr@kn 760 \def\MT@abbr@kn@c{kerning codes}
\MT@abbr@kn@c 761 \def\MT@abbr@kn@inh{kerning inheritance}
762 \def\MT@abbr@tr{tracking}
\MT@abbr@kn@inh 763 \def\MT@abbr@tr@c{tracking amount}
\MT@abbr@tr
\MT@rba@protrusion
\MT@rba@expansion 764 \def\MT@rba@protrusion{pr}
\MT@rba@spacing 765 \def\MT@rba@expansion{ex}
766 \def\MT@rba@spacing{sp}
\MT@rba@kerning 767 \def\MT@rba@kerning{kn}
\MT@rba@tracking 768 \def\MT@rba@tracking{tr}

```

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

```

\MT@features@long 769 \def\MT@features{pr,ex,sp,kn,tr}
770 \def\MT@features@long{protrusion,expansion,spacing,kerning,tracking}

```

`\MT@is@feature` Whenever an optional argument accepts a list of features, we can use this command to check whether a feature exists in order to prevent a rather confusing 'Missing \endcsname inserted' error message. The feature (long form) must be in `<#1>`, the type of list to ignore in `<#2>`, then comes the action.

```

771 \def\MT@is@feature#1#2{%
772 \MT@in@clist{#1}\MT@features@long
773 \ifMT@inlist@
774 \expandafter\@firstofone
775 \else
776 \MT@error{`#1' is not an available micro-typographic\MessageBreak
777 feature. Ignoring #2}{Available features are: ` \MT@features@long'.}%
778 \expandafter\@gobble
779 \fi
780 }

```

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.

```
781 \@ifl@aded{tex}{wordcount}{%
782   \MT@warning@nl{Detected the `wordcount' utility.\MessageBreak
783     Disabling `MT@MT', since it wouldn't work}%
784   \MT@clear@options\endinput}\relax
```

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

```
785 \@ifclassloaded{minimal}{%
786   \MT@warning@nl{Detected the `minimal' class.\MessageBreak
787     Expect lots of warnings and some malfunctions.\MessageBreak
788     You might want to use a proper class instead}%
789 }\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`.

```
790 </package>
791 <*package|letterspace>
792 <plain>\MT@requires@latexl{
793 \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.

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

Don't hesitate with `miniltx`.

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

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

```
796 \def\MT@with@package@T#1{\@ifpackageloaded{#1}\@firstofone\@gobble}
797 </package|letterspace>
798 <*package>
```

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

```
799 \def\MT@with@babel@and@T#1{%
800   \MT@ifdefined@n@T{opt@babel.\@pkgextension}{%
801     \@expandtwoargs\MT@inclist{#1}
802     {\csname opt@babel.\@pkgextension\endcsname,\@classoptionslist}%
803     \ifMT@inlist@expandafter\@gobble\fi
804   }\@gobble
805 }
```

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

Peter Wilson was so kind to provide the `\l@dunhbox@line` hook in `ledmac` to allow for protrusion. `\leftmargin@kern` and `\rightmargin@kern` are new primitives

of pdf \TeX 1.21b (aka. 1.30.0). They are also part of recent X \TeX . The successor packages eledmac and reledmac are also supported.

```

806 /package
807 /pdf $\TeX$ -def\MT@requires@pdftex5{
808 (*pdf $\TeX$ -def|luatex-def|xetex-def)
809 \def\MT@ledmac@setup{%
810 \ifMT@protrusion
811 \MT@ifdefined@c@TF\l@dunhbox@line{%
\MT@led@unhbox@line Hook.
812 \MT@info@n1{Patching ((r)e)ledmac to enable character protrusion}%
813 \let\MT@led@unhbox@line\l@dunhbox@line
814 \renewcommand*{\l@dunhbox@line}[1]{%
815 \ifhbox##1%
816 \kern\leftmarginkern##1%
817 \expandafter\MT@led@unhbox@line\expandafter##1\expandafter
818 \kern\rightmarginkern##1%
819 \fi
820 }%
821 }{%
822 \MT@warning@n1{%
823 Character protrusion in paragraphs with line\MessageBreak
824 numbering will only work if you update ledmac,\MessageBreak
825 or use one of its successors, eledmac or reledmac}%
826 }%
827 \fi
828 }
829 /pdf $\TeX$ -def|luatex-def|xetex-def)
830 (*pdf $\TeX$ -def)
831 }{
832 \def\MT@ledmac@setup{%
833 \ifMT@protrusion
834 \MT@warning@n1{%
835 The pdf $\TeX$  version you are using does not allow\MessageBreak
836 character protrusion in paragraphs with line\MessageBreak
837 numbering by the `((r)e)ledmac' package.\MessageBreak
838 Upgrade pdf $\TeX$  to version 1.30 or later}%
839 \fi
840 }
841 }
842 /pdf $\TeX$ -def

```

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

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

```

843 (*package|letterspace)
844 (*package)
845 \def\MT@restore@p@h{\chardef\%` \% \chardef\#` \# }

```

\ifMT@unicode Two new conditionals for use with X \TeX or Lua \TeX .

```

\ifMT@fontspec 846 \newif\ifMT@unicode
847 \MT@with@package@T{xunicode}\MT@unicodetrue
848 \newif\ifMT@fontspec
849 \MT@with@package@T{fontspec}\MT@fontspectrue

```

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

\MT@tikz@setup

```

850 \let\MT@maybe@gobble@with@tikz@firstofone
851 \def\MT@tikz@setup{%
852 \def\MT@maybe@gobble@with@tikz{%
853 \ifnum\tikz@expandcount>\z@

```

```

854     \expandafter\@gobble
855     \else
856     \expandafter\@firstofone
857     \fi}}

```

`\MT@setupfont@hook`

This hook will be executed every time a font is set up (inside a group).

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

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

```

858 \def\MT@setupfont@hook{%

```

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

```

859   \MT@if@false
860   \MT@with@babel@and@T{spanish} \MT@if@true
861   \MT@with@babel@and@T{galician} \MT@if@true
862   \MT@with@babel@and@T{mexican} \MT@if@true
863   \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.)

```

864   \MT@with@package@T{csquotes}{%
865     \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`.

```

866   \MT@if@false
867   \MT@with@package@T{hyperref} \MT@if@true
868   \MT@with@package@T{tex4ht} \MT@if@true
869   \MT@with@package@T{mathastext} \MT@if@true
870   \ifMT@if@MT@restore@p@h\fi
871   \MT@with@package@T{tikz} \MT@tikz@setup
872 }

```

Check again at the end of the preamble.

```

873 </package>
874 \MT@addto@setup{%
875 <*package>

```

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

```

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

```

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

```

888   \MT@gl@et\MT@setupfont@hook\@empty
889   \MT@if@false
890   \MT@with@babel@and@T{spanish} \MT@if@true

```

```

891 \MT@with@babel@and@T{galician}\MT@if@true
892 \MT@with@babel@and@T{mexican} \MT@if@true
893 \ifMT@if@
894 \g@addto@macro\MT@setupfont@hook{%
895 \MT@ifdefined@c@T\percentsign{\let%\percentsign}}%
896 \fi
897 \MT@with@package@T{csquotes}{%
898 \ifpackage@later{csquotes}{2005/05/11}{%
899 \g@addto@macro\MT@setupfont@hook\@disablequotes
900 }{%
901 \MT@warning@n1{%
902 Should you receive warnings about unknown slot\MessageBreak
903 numbers, try upgrading the `csquotes' package}%
904 }%
905 }%

```

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.

```

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

```

The `listings` package makes numbers and letters active,

```

926 \MT@with@package@T{listings}{%
927 \g@addto@macro\MT@cfg@catcodes{%
928 \MT@while@num{"30}{ "3A}{\catcode\@tempcnta=12\relax}%
929 \MT@while@num{"41}{ "5B}{\catcode\@tempcnta=11\relax}%
930 \MT@while@num{"61}{ "7B}{\catcode\@tempcnta=11\relax}%
931 }%

```

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

```

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

```

Inside a listing, `\space` is redefined.

```

934 \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.

```

935 \let\lst@ProcessLetter\@empty
936 }%
937 }%

```

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. Also, we have to disable expansion within `soul`'s trial run. Under plain `TEX`, `soul` doesn't register itself the `LATEX` way, so we just test for its main command.

```

938 /package
939 \ifx\SOULE@\undefined\else
940 \soulregister\lsstyle 0%
941 \soulregister\textls 1%
942 \ifx\XeTeXrevision@\undefined
943 \let\MT@SOULE@doword\SOULE@doword
944 \def\SOULE@doword{\pdfadjustspacing=\z@ \MT@SOULE@doword}%
945 \fi
946 \fi
947 *package
948 \MT@with@package@T{tikz}\MT@tikz@setup

```

Compatibility with the `pinyin` package (from CJK): disable `microtyp` 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.

```

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

```

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

```

966 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`).

```

967 *pdftex-def|xetex-def|luatex-def
968 \def\MT@setupfont{%

```

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

```

969 xetex-def|luatex-def \MT@font

```

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

```

970 \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).

```

971 pdftex-def\MT@requires@pdftex7{
972 pdftex-def|luatex-def\g@addto@macro\MT@setupfont\MT@copy@font
973 pdftex-def}\relax

```

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


```
974 \g@addto@macro\MT@setupfont{%
975   \MT@exp@two@c\MT@split@name\string\MT@font/\@nil
```

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

```
976   \MT@exp@one@n\MT@find@file\MT@family
977   \ifx\MT@familyalias\@empty \else
978     \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.)

```
979 % \ifx\cf@encoding\cf@encoding\else\@enc@update\fi
980 }
```

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

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

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

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

```
991   \MT@protrusion
992 <pdfTeX-def|luatex-def> \MT@expansion
993 }
```

Interword spacing and kerning (pdfTeX 1.40).

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

Disable ligatures (pdfTeX 1.30).

```
999 <pdfTeX-def>\MT@requires@pdfTeX5{
1000 <pdfTeX-def|luatex-def>\g@addto@macro\MT@setupfont\MT@noLigatures
1001 <pdfTeX-def>}\relax
1002 \g@addto@macro\MT@setupfont{%
```

Debugging.

```
1003 <debug>\MT@show@pdfannot1%
```

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

```
1004   \MT@register@font
1005   \fi
1006 }
1007 </pdfTeX-def|xetex-def|luatex-def>
```

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

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

or `\SetExpansion` in the preamble, or when the package has been loaded with the `copyfonts` option.

```
1008 (*pdftex-def|luatex-def)
1009 \let\MT@copy@font\relax
1010 (pdftex-def)\MT@requires@pdftex{
1011 \def\MT@copy@font{%
```

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

```
1012 \xdef\MT@font@copy{\csname\MT@font/\MT@prcontext/\MT@excontext\endcsname}%
1013 \expandafter\ifx\MT@font@copy\relax
```

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

```
1014 \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%
1015 \expandafter\ifx\MT@font@orig\relax
1016 \MT@exp@two@{c\MT@gl@t\MT@font@orig\font@name
1017 \else
1018 \MT@exp@two@{let\font@name\MT@font@orig
1019 \fi
1020 (pdftex-def) \global\MT@exp@two@{pdfcopyfont\MT@font@copy\font@name
```

Even though Lua_TE_X also provides the primitive from pdf_TE_X (even renamed to `\copyfont`, that is, 'promoted' as per the Lua_TE_X manual), it is seriously crippled in that OpenType features will be lost. Therefore, we do not copy the font but load it anew.

```
1021 (luatex-def) \MT@exp@two@{c\MT@lua@copyfont\meaning\font@name\@nil
1022 (debug)\MT@dinfo{creating new copy: \MT@font@copy}%
```

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

```
1023 \MT@map@clist@{c\MT@active@features{%
1024 \MT@exp@cs\ifx\MT@\@nameuse\{MT@abbr@##1}\relax\else
1025 \def\@tempa{##1}%
1026 \MT@exp@cs\MT@map@tlist@{c\MT@##1@doc@contexts}\MT@rem@from@list
1027 \fi
1028 }%
1029 \fi
1030 \MT@exp@two@{let\MT@font\MT@font@copy
```

We only need the font identifier for letterspacing.

```
1031 \let\font@name\MT@font@copy
```

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

```
1032 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
1033 }
```

`\MT@rem@from@list`

```
1034 \def\MT@rem@from@list#1{%
1035 \MT@exp@cs\ifx\MT@\@tempa @#1font@list}\relax\else
1036 \expandafter\MT@exp@one@\expandafter\MT@rem@from@clist\expandafter
1037 \MT@font \csname MT@\@tempa @#1font@list\endcsname
1038 \fi
1039 }
1040 (pdftex-def)\relax
```

`\MT@lua@copy@font` `<#1>` and `<#2>` are 'select' and 'font', respectively, `<#3>` is the font spec.

```
1041 (luatex-def)\def\MT@lua@copyfont #1 #2 #3\@nil{%
1042 (luatex-def) \global\expandafter\font\MT@font@copy=#3\relax}
1043 (pdftex-def|luatex-def)
```

Here's the promised dirty trick for users of older pdf_TE_X 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@fix@fontdimen@six`
`\MT@dimen@six`

If `\fontdimen 6` is zero, character protrusion, spacing, kerning and tracking won't work, and we could skip the settings (for example, the `dsfont` fonts don't specify this dimension; this is probably a bug – the `fourier` and `newpx/newtx` packages have been fixed in the meantime). However, we can fix it ourselves – only tracking still doesn't work (it seems that `\letterspacefont` uses the `\fontdimen 6` from the original font). `XYTEX` doesn't provide an equivalent to `\pdffontsize`, so we use the nominal size instead.

```
1044 <{*pdftex-def|luatex-def|xetex-def}
1045 \def\MT@fix@fontdimen@six{%
1046   \ifnum\fontdimen6\MT@font=\z@
1047     \fontdimen6\MT@font=%
1048   <{pdftex-def} \pdffontsize\MT@font
1049   <{luatex-def} \MT@requires@luatex4{\pdffeedback fontsize}{\pdffontsize}\MT@font
1050   <{xetex-def} \MT@size pt
1051   \MT@info{Fixing zero \string\fontdimen 6 for font `'\MT@font'\MessageBreak
1052     (new value: \the\fontdimen6\MT@font)}%
1053   \MT@glet@nc{\MT@@font-fake6}\empty
1054   \fi
1055   \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1056 }
1057 </pdftex-def|luatex-def|xetex-def}
```

`\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@series 1058 <{*package}
\MT@shape 1059 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
1060   \def\MT@encoding{#1}%
\MT@size 1061   \ifMT@fontspec
1062     \edef\MT@family{\MT@scrubfeature#2()\relax}%
1063   \else
1064     \def\MT@family{#2}%
1065   \fi
1066   \def\MT@series {#3}%
1067   \def\MT@shape {#4}%
1068   \def\MT@size {#5}%
1069   \MT@fix@fontdimen@six
```

`\MT@familyalias` Alias family?

```

1070 \MT@ifdefined@n@TF{MT@MT@family @alias}%
1071 {\MT@let@cn\MT@familyalias{MT@MT@family @alias}}%
1072 {\let\MT@familyalias\empty}%
1073 }

\MT@scrubfeature Remove one resp. all feature counters (fontspec).
\MT@scrubfeatures 1074 \def\MT@scrubfeature#1(#2)#3\relax{#1}
1075 \def\MT@scrubfeatures#1(#2)#3\relax{%
1076 #1%
1077 \ifx\relax#3\relax\else
1078 \MT@scrubfeatures#3\relax
1079 \fi
1080 }

\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 1081 \newif\ifMT@do
1082 \def\MT@maybe@do#1{%
    (but only if the feature isn't globally set to false)
1083 \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).

1084 \MT@dotrue
1085 \edef\@tempa{\csname MT@#1@setname\endcsname}%
1086 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
1087 \MT@ifdefined@n@TF{MT@checklist@#1}%
1088 {\csname MT@checklist@#1\endcsname}%
1089 {\MT@checklist@{#1}}%
1090 {#1}%
1091 }%
1092 \else
1093 \MT@dofalse
1094 \fi
1095 \ifMT@do

    \MT@feat stores the current feature.

1096 \def\MT@feat{#1}%
1097 \csname MT@set@#1@codes\endcsname
1098 \else
1099 \MT@ifstreq{#1}{tr}%
1100 {\let\MT@info@nottracking\MT@info@nottracking}%
1101 {\MT@vinfo{... No \@nameuse{MT@abbr@#1}}}%
1102 \fi
1103 }

\MT@info@nottracking To defer the message to after the font has actually been logged.
\MT@info@nottracking@ 1104 \let\MT@info@nottracking\relax
1105 \def\MT@info@nottracking@{\MT@vinfo{... No tracking}}

\MT@dinfo@list
1106 <debug>\def\MT@dinfo@list#1#2#3{\MT@dinfo@n1}{1}{\@nameuse{MT@abbr@#1}: #2
1107 <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).
1108 \def\MT@checklist@#1#2{%
1109 <!debug> \MT@ifdefined@n@T
1110 <debug> \MT@ifdefined@n@TF
1111 {MT@#2list@#1@\@tempa}{%

    Begin a (neatly masqueraded) \expandafter orgy to test whether the font attribute

```

is in the list.

```

1112 \expandafter\MT@exp@one@n\expandafter\MT@in@clist
1113 \csname MT@#1\expandafter\endcsname
1114 \csname MT@#2list@#1@\tempa\endcsname
1115 \ifMT@inlist@
1116 <debug>\MT@info@list{#2}{#1}{in}%
1117 \MT@dotrue
1118 \else
1119 <debug>\MT@info@list{#2}{#1}{not in}%
1120 \MT@dofalse
1121 \expandafter\MT@clist@break
1122 \fi
1123 }%

```

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.

```

1124 <debug> {\MT@info@list{#2}{#1}{}}%
1125 }

```

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

```

1126 \def\MT@checklist@family#1{%
1127 <!debug> \MT@ifdefined@n@T
1128 <debug> \MT@ifdefined@n@TF
1129 {MT@#1list@family@\tempa}%
1130 \MT@exp@two@n\MT@in@clist
1131 \MT@family{\csname MT@#1list@family@\tempa\endcsname}%
1132 \ifMT@inlist@
1133 <debug>\MT@info@list{#1}{family}{in}%
1134 \MT@dotrue
1135 \else
1136 <debug>\MT@info@list{#1}{family}{not in}%
1137 \MT@dofalse
1138 \ifx\MT@familyalias\empty \else
1139 \MT@exp@two@n\MT@in@clist
1140 \MT@familyalias{\csname MT@#1list@family@\tempa\endcsname}%
1141 \ifMT@inlist@
1142 <debug> \MT@info@list{#1}{family alias}{in}%
1143 \MT@dotrue
1144 <debug>\else\MT@info@list{#1}{family alias}{not in}%
1145 \fi
1146 \fi
1147 \fi
1148 \ifMT@do \else
1149 \expandafter\MT@clist@break
1150 \fi
1151 }%
1152 <debug> {\MT@info@list{#1}{family}{}}%
1153 }

```

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

```

1154 \def\MT@checklist@size#1{%
1155 <!debug> \MT@ifdefined@n@T
1156 <debug> \MT@ifdefined@n@TF
1157 {MT@#1list@size@\tempa}%
1158 \MT@exp@cs\MT@in@rlist{MT@#1list@size@\tempa}%
1159 \ifMT@inlist@
1160 <debug>\MT@info@list{#1}{size}{in}%
1161 \MT@dotrue
1162 \else
1163 <debug>\MT@info@list{#1}{size}{not in}%
1164 \MT@dofalse
1165 \expandafter\MT@clist@break
1166 \fi
1167 }%

```

```
1168 <debug> {\MT@info@list{#1}{size}}}%
1169 }
```

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

```
1170 \def\Mt@checklist@font#1{%
1171 <!debug> \MT@ifdefined@n@T
1172 <debug> \MT@ifdefined@n@TF
1173     {\MT#1list@font@\@tempa}{%
```

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

```
1174     \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1175     \expandafter\Mt@expone@n\expandafter\Mt@in@clist\expandafter
1176     \@tempb \csname Mt@#1list@font@\@tempa\endcsname
1177     \ifMT@inlist@
1178 <debug>\MT@info@list{#1}{font}{in}%
1179     \expandafter\Mt@clist@break
1180     \else
1181 <debug>\MT@info@list{#1}{font}{not in}%
1182     \MT@dofalse
1183     \fi
1184     }%
1185 <debug> {\MT@info@list{#1}{font}}}%
1186 }
```

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.

```
1187 \newif\ifMT@nofamily
1188 </package>
```

\MT@protrusion Set up for protrusion?

```
1189 <*pdfTeX-def|xetex-def|luatex-def>
1190 \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.

```
1191 \def\Mt@set@pr@codes{%
1192     \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).

```
1193     \MT@if@list@exists{%
1194         \ifMT@nofamily
1195             \MT@ifdefined@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1196                 \MT@info@n1{Loading generic protrusion settings for font family\MessageBreak
1197                     ~\MT@family' (encoding: \MT@encoding).\MessageBreak
1198                     For optimal results, create family-specific settings.\MessageBreak
1199                     See the microtype manual for details}%
1200                 \MT@glet@nc{\MT@encoding-\MT@family-settings}\@empty
1201             }%
1202         \fi
1203         \MT@get@opt
1204         \MT@reset@pr@codes
```

Get the name of the inheritance list and parse it.

```
1205     \MT@get@inh@list
```

Set an input encoding?

```
1206     \MT@set@inputenc{c}%
```

Load additional lists?

```
1207 \MT@load@list\MT@pr@c@name
1208 \MT@set@listname
```

Load the main list.

```
1209 \MT@let@cn@tempc{MT@pr@c@MT@pr@c@name}%
1210 \expandafter\MT@set@codes\@tempc,\relax,%
1211 } \MT@reset@pr@codes
1212 }
```

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

```
1213 \def\MT@set@all@pr#1#2{%
1214 debug\MT@dinfol{3}{-- lp/rp: setting all to #1/#2}%
1215 \let\MT@temp\@empty
1216 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
1217 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
1218 \MT@do@font\MT@temp
1219 }
```

\MT@reset@pr@codes@ All protrusion codes are zero for new fonts. However, if we have to reload the font due to different contexts, we have to reset them. This command will be changed by \microtypecontext if necessary.

```
1220 \def\MT@reset@pr@codes@{\MT@set@all@pr\z@z@}
1221 \let\MT@reset@pr@codes\relax
```

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

```
1222 \def\MT@the@pr@code{\@tempcntb}
1223 pdfTeX-def|luatex-def
1224 pdfTeX-def\MT@requires@pdfTeX6
1225 luatex-def\MT@requires@luatex3
1226 {\def\MT@the@pr@code@tr{%
1227 \numexpr\@tempcntb+\MT@letterspace@/2\relax
1228 }
1229 }\relax
1230 pdfTeX-def|luatex-def
```

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

```
1231 \def\MT@set@codes#1,{%
1232 \ifx\relax#1\@empty\else
1233 \MT@split@codes #1==\relax
1234 \expandafter\MT@set@codes
1235 \fi
1236 }
```

\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.

```
1237 \def\MT@split@codes#1=#2=#3\relax{%
1238 \def\@tempa{#1}%
1239 \ifx\@tempa\@empty \else
1240 \MT@get@slot
1241 pdfTeX-def|luatex-def \ifnum\MT@char > \m@ne
1242 xetex-def \ifx\MT@char\@empty \else
1243 \MT@get@char@unit
1244 \csname MT@\MT@feat @split@val\endcsname#2\relax
1245 \fi
1246 \fi
1247 }
```

\MT@pr@split@val

```
1248 \def\MT@pr@split@val#1,#2\relax{%
1249 \def\@tempb{#1}%
```

```

1250 \MT@ifempty\@tempb\relax{%
1251 \MT@scale@to@em
1252 \lrcode\MT@font\MT@char=\MT@the@pr@code
1253 <debug>\MT@info@n1{4}{;;; lp (\MT@char): \number\lrcode\MT@font\MT@char\space: [#1]}%
1254 }%
1255 \def\@tempb{#2}%
1256 \MT@ifempty\@tempb\relax{%
1257 \MT@scale@to@em
1258 \rprcode\MT@font\MT@char=\MT@the@pr@code
1259 <debug>\MT@info@n1{4}{;;; rp (\MT@char): \number\rprcode\MT@font\MT@char\space: [#2]}%
1260 }%

```

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

```

1261 \MT@ifdefined@c@T\MT@pr@inh@name{%
1262 \MT@ifdefined@nT{\MT@inh@\MT@pr@inh@name @\MT@char @}{%
1263 \MT@exp@cs\MT@map@tlist@c
1264 {MT@inh@\MT@pr@inh@name @\MT@char @}%
1265 \MT@set@pr@heirs
1266 }%
1267 }%
1268 }

```

`\MT@scale@to@em` Since pdfTeX version 0.14h, we have to adjust the protrusion factors (i.e., convert numbers from thousandths of character width to thousandths of an em of the font). We have to do this *before* setting the inheriting characters, so that the latter inherit the absolute value, not the relative one if they have a differing width (e.g., the ‘ff’ ligature). Unlike `protcode.tex` and `pdfcprot`, we do not calculate with `\lrcode` resp. `\rprcode`, since this would disallow protrusion factors larger than the character width (since `\[lr]rcode`’s limit is 1000). Now, the maximum protrusion is 1 em of the font.

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

```

1269 <pdfTeX-def>\MT@requires@pdfTeX3{
1270 \def\MT@scale@to@em{%
1271 \@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`.

```

1272 \MT@scale\@tempcntb \@tempb \MT@dimen@six
1273 \ifnum\@tempcntb=<z@ \else
1274 \MT@scale@factor
1275 \fi
1276 }

```

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

```

1277 \def\MT@get@charwd{%
1278 <pdfTeX-def>
1279 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1280 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1281 ^^Q \MT@count=\wd\z@
1282 </pdfTeX-def>
1283 <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).

```

1284 <xetex-def>
1285 \ifnum\MT@char@<z@
1286 \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%

```



```

1287 \MT@count=\wd\z@
1288 \else
1289 \MT@count=\fontcharwd\MT@font\MT@char@\relax
1290 \fi
1291 /xetex-def
1292 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1293 }

```

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`.

```

1294 (*pdfTeX-def)
1295 \MT@requires@pdfTeX6{
1296 \g@addto@macro\MT@get@charwd{%
1297 \MT@ifdefined@cT\MT@letterspace@
1298 {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1299 }
1300 }\relax
1301 }{

```

No adjustment with versions 0.14f and 0.14g.

```

1302 \def\MT@scale@to@em{%
1303 \MT@count=\@tempb\relax
1304 \ifnum\MT@count=\z@ \else
1305 \MT@scale@factor
1306 \fi
1307 }

```

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

```

1308 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1309 }
1310 /pdfTeX-def
1311 /pdfTeX-def|xetex-def|luatex-def

```

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

```

1312 (*package)
1313 \def\MT@get@font@dimen#1{%
1314 \ifnum\fontdimen#1\MT@font=\z@
1315 \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1316 \@backslashchar fontdimen #1 (it's zero)! \MessageBreak
1317 You should use a different `unit' for \MT@curr@list@name}%
1318 \else
1319 \MT@count=\fontdimen#1\MT@font
1320 \fi
1321 }

```

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

```

1322 \def\MT@info@missing@char{%
1323 \MT@info@n1{Character '\the\MT@toks'
1324 ^^X \ifnum\MT@char<\z@ is missing\else
1325 ^^X \iffontchar\MT@font\MT@char@
1326 has a width of 0pt
1327 ^^X \else is missing\fi\fi
1328 ^^Q \MessageBreak (it's probably missing)
1329 \MessageBreak in font '\MT@font'. \MessageBreak
1330 Ignoring protrusion settings for this character}%
1331 }

```

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

```

1332 \def\MT@scale@factor{%
1333 \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1334 \expandafter\MT@scale\expandafter \@tempcntb
1335 \csname MT@\MT@feat @factor@\endcsname \@m
1336 \fi

```

```

1337 \ifnum\@tempcntb>\csname MT@\MT@feat @max\endcsname\relax
1338 \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @max}%
1339 \else
1340 \ifnum\@tempcntb<\csname MT@\MT@feat @min\endcsname\relax
1341 \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @min}%
1342 \fi
1343 \fi
1344 }

```

`\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.

```

1345 \def\MT@warn@code@too@large#1{%
1346 \@tempcnta=#1\relax
1347 \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1348 \expandafter\MT@scale\expandafter\@tempcnta\expandafter
1349 \@m \csname MT@\MT@feat @factor@\endcsname
1350 \fi
1351 \MT@scale\@tempcnta \MT@dimen@six \MT@count
1352 \MT@warning@n1{The \@nameuse{MT@abbr@\MT@feat} code \@tempb\space
1353 is too large for character\MessageBreak
1354 \the\MT@toks' in \MT@curr@list@name.\MessageBreak
1355 Setting it to the maximum of \number\@tempcnta}%
1356 \@tempcntb=#1\relax
1357 }

```

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

```

1358 \def\MT@get@opt{%
1359 \MT@set@listname

```

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

```

\MT@sp@factor@ 1360 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}{%
\MT@kn@factor@ 1361 \MT@let@nn{MT@\MT@feat @factor@}
1362 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}%
1363 \MT@vinfo{... : Multiplying \@nameuse{MT@abbr@\MT@feat} codes by
1364 \number\csname MT@\MT@feat @factor@\endcsname/1000}%
1365 }{%
1366 \MT@let@nn{MT@\MT@feat @factor@}{MT@\MT@feat @factor}%
1367 }%

```

`\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@ 1368 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}{%
1369 \MT@let@nn{MT@\MT@feat @unit@}%
1370 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}%
1371 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\@empty
1372 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1373 relative to character widths}%
1374 \else
1375 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\m@ne
1376 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1377 relative to width of space}%
1378 \fi
1379 \fi
1380 }{%
1381 \MT@let@nn{MT@\MT@feat @unit@}{MT@\MT@feat @unit}%
1382 }%

```

`\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.

```

1383 \let\MT@get@char@unit\relax
1384 \let\MT@get@space@unit\@gobble
1385 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\@empty
1386 \let\MT@get@char@unit\MT@get@charwd
1387 \else
1388 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\m@ne
1389 \let\MT@get@space@unit\MT@get@font@dimen
1390 \else
1391 \MT@exp@cs\MT@get@unit{MT@\MT@feat @unit@}%
1392 \fi
1393 \fi

```

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

```

1394 \MT@ifdefined@n@T{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @preset}{%
1395 \csname MT@preset@\MT@feat\endcsname
1396 \MT@let@nc{MT@reset@\MT@feat @codes}\relax
1397 }%
1398 }

```

`\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.

```

1399 \def\MT@get@unit#1{%
1400 \expandafter\MT@get@unit@#1 e!\@nil
1401 \ifx\x\@empty\else\let#1x\fi
1402 \@defaultunits\@tempdima#1 pt\relax\@nnil
1403 \ifdim\@tempdima=\z@
1404 \MT@warning@n1{%
1405 Cannot set \@nameuse{MT@abbr@\MT@feat} factors relative to zero\MessageBreak
1406 width. Setting factors of list \@nameuse{MT@\MT@feat @c@name}'\MessageBreak
1407 relative to character widths instead}%
1408 \let#1\@empty
1409 \let\MT@get@char@unit\MT@get@charwd
1410 \else
1411 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} factors relative
1412 to \the\@tempdima}%
1413 \MT@count=\@tempdima\relax
1414 \fi
1415 }
1416 \def\MT@get@unit@#1e#2#3\@nil{%
1417 \ifx\#3\@empty\else
1418 \if m#2%
1419 \edef\x{#1\fontdimen6\MT@font}%
1420 \else
1421 \if x#2%
1422 \edef\x{#1\fontdimen5\MT@font}%
1423 \fi
1424 \fi
1425 \fi
1426 }

```

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

```
1427 \def\MT@set@inputenc#1{%
```

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

```

1428 \def\MT@cat{#1}%
1429 \edef\@tempa{MT@\MT@feat @#1\csname MT@\MT@feat @#1@name\endcsname @inputenc}%
1430 \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1431 }

```

`\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.

```

1432 \MT@addto@setup{%
1433   \ifpackageloaded{inputenc}{%
1434     \@ifpackagelater{inputenc}{2006/02/22}{%
1435       \def\MT@set@inputenc@{%
1436         \MT@ifstreq\inputencodingname{\csname\@tempa\endcsname}\relax
1437         \MT@load@inputenc
1438       }%
1439     }%
1440     \let\MT@set@inputenc@\MT@load@inputenc
1441   }%
1442 }%
1443 \def\MT@set@inputenc@{%
1444   \MT@warning@n1{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1445     \MessageBreak package isn't loaded. Ignoring input encoding}%
1446 }%
1447 }%
1448 }

```

`\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.

```

1449 \def\MT@load@inputenc{%
1450   \MT@cfg@catcodes
1451   (debug)\MT@dinfo@n1{1}{loading input encoding: \@nameuse{\@tempa}}%
1452   \inputencoding{\@nameuse{\@tempa}}%
1453 }
1454 (/package)

```

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

```

1455 (*pdfTeX-def|xetex-def|luatex-def)
1456 \def\MT@set@pr@heirs#1{%
1457   \lcode\MT@font #1=\lcode\MT@font\MT@char\relax
1458   \rcode\MT@font #1=\rcode\MT@font\MT@char\relax
1459   (debug)\MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
1460   (debug)\MT@dinfo@n1{4}{;;; lp/rp (#1): \number\lcode\MT@font\MT@char\space/%
1461   (debug) \number\rcode\MT@font\MT@char\space}%
1462 }

```

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

```

\MT@preset@pr@ 1463 \def\MT@preset@pr{%
1464   \expandafter\expandafter\expandafter\MT@preset@pr@
1465   \csname MT@pr@c@\MT@pr@c@name @preset\endcsname\@nil
1466 }
1467 \def\MT@preset@pr@#1,#2\@nil{%
1468   \ifx\MT@pr@unit@\empty
1469     \MT@warn@preset@twidth{pr}%
1470     \let\MT@preset@aux\MT@preset@aux@factor
1471   \else
1472     \def\MT@preset@aux{\MT@preset@aux@space2}%
1473   \fi
1474   \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
1475   \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
1476   \MT@set@all@pr@\@tempa\@tempb
1477 }

```

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

```

\MT@preset@aux@factor 1478 \def\MT@preset@aux@factor#1#2{%
\MT@preset@aux@space 1479   \@tempcntb=#1\relax
1480   \MT@scale@factor
1481   \edef#2{\number\@tempcntb}%
1482 }
1483 \def\MT@preset@aux@space#1#2#3{%
1484   \def\@tempb{#2}%
1485   \MT@get@space@unit#1%
1486   \MT@scale@to@em
1487   \edef#3{\number\@tempcntb}%

```

```

1488 }
\MT@warn@preset@tewidth
1489 \def\MT@warn@preset@tewidth#1{%
1490   \MT@warning@n1{%
1491     Cannot preset characters relative to their widths\MessageBreak
1492     for \@nameuse{MT@abbr#1} list \@nameuse{MT@#1@c@name}'. Presetting them%
1493     \MessageBreak relative to lem instead}%
1494 }
1495 (/pdftex-def|xetex-def|luatex-def)

```

14.2.2 Expansion

`\MT@expansion` Set up for expansion?

```

1496 (*pdftex-def|luatex-def)
1497 \def\MT@expansion{\MT@maybe@do{ex}}

```

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

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

```

1498 \def\MT@set@ex@codes@s{%
1499   \MT@if@list@exists{%
1500     \MT@get@ex@opt
1501     \let\MT@get@char@unit\relax
1502     \MT@reset@ef@codes
1503     \MT@get@inh@list
1504     \MT@set@inputenc{c}%
1505     \MT@load@list\MT@ex@c@name
1506     \MT@set@listname
1507     \MT@let@cn\@tempc{MT@ex@c@\MT@ex@c@name}%
1508     \expandafter\MT@set@codes\@tempc,\relax,%
1509     \MT@expandfont
1510   }\relax
1511 }
1512 (/pdftex-def|luatex-def)

```

`\MT@set@ex@codes@n` If, on the other hand, all characters should be expanded by the same amount, we only take the first optional argument to `\SetExpansion` into account.

`\ifMT@nonselected` We need this boolean in `\MT@if@list@exists` so that no warning for missing lists will be issued.

```

1513 (package)\newif\ifMT@nonselected
1514 (*pdftex-def|luatex-def)
1515 \def\MT@set@ex@codes@n{%
1516   \MT@nonselectedtrue
1517   \MT@if@list@exists
1518   \MT@get@ex@opt
1519   {%
1520     \let\MT@stretch@ \MT@stretch
1521     \let\MT@shrink@ \MT@shrink
1522     \let\MT@step@ \MT@step
1523     \let\MT@auto@ \MT@auto
1524     \let\MT@ex@factor@\MT@ex@factor
1525   }%
1526   \MT@reset@ef@codes
1527   \MT@expandfont
1528   \MT@nonselectedfalse
1529 }

```

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

```

1530 \let\MT@set@ex@codes\MT@set@ex@codes@n

```

`\MT@expandfont` Expand the font. For some reason, older LuaTeX versions freeze if the `autoexpand` modifier is missing. Can't be bothered to find out why. For newer versions, we could also use the function `font.setexpansion`, or, in the future, `luaotfload's expansion font` feature.

```

1531 (*luatex-def)
1532 \MT@requires@luatex3{
1533 \MT@requires@luatex4{\let\pdffontexpand\expandglyphsinfont}\relax
1534 \ifnum\luatexversion<79
1535 \def\MT@expandfont{%
1536 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ autoexpand\relax
1537 }
1538 \else
1539 \def\MT@expandfont{%
1540 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@\relax
1541 }
1542 \fi
1543 }{
1544 (/luatex-def)
1545 \def\MT@expandfont{%
1546 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
1547 }
1548 (luatex-def)

```

`\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).

```

1549 \def\MT@set@all@ex#1{%
1550 (debug)\MT@dinfol{3}{-- ex: setting all to \number#1}%
1551 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
1552 }
1553 \def\MT@reset@ef@codes@{\MT@set@all@ex\MT@ex@factor@}

```

`\MT@reset@ef@codes` However, this is only necessary for pdfTeX versions prior to 1.20, or LuaTeX < 0.90 (actually, I think, 0.87).

```

1554 (pdfTeX-def)\MT@requires@pdfTeX4
1555 (luatex-def)\MT@requires@luatex5
1556 {
1557 \def\MT@reset@ef@codes{%
1558 \ifnum\MT@ex@factor@=\@m \else
1559 \MT@reset@ef@codes@
1560 \fi
1561 }
1562 }{
1563 \let\MT@reset@ef@codes\MT@reset@ef@codes@
1564 }

```

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

```

1565 \def\MT@ex@split@val#1\relax{%
1566 \@tempcntb=#1\relax

```

Take an optional factor into account.

```

1567 \ifnum\MT@ex@factor@=\@m \else
1568 \MT@scale\@tempcntb \MT@ex@factor@ \@m
1569 \fi
1570 \ifnum\@tempcntb > \MT@ex@max
1571 \MT@warn@ex@too@large\MT@ex@max
1572 \else
1573 \ifnum\@tempcntb < \MT@ex@min
1574 \MT@warn@ex@too@large\MT@ex@min
1575 \fi
1576 \fi
1577 \efcode\MT@font\MT@char=\@tempcntb
1578 (debug)\MT@dinfol{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

```

Heirs, heirs, I love thy heirs.

```

1579 \MT@ifdefined@c@T\MT@ex@inh@name{%
1580   \MT@ifdefined@n@T{MT@inh@MT@ex@inh@name @\MT@char @}{%
1581     \MT@exp@cs\MT@map@tlist@c{MT@inh@MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
1582   }%
1583 }%
1584 }

```

\MT@warn@ex@too@large

```

1585 \def\MT@warn@ex@too@large#1{%
1586   \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for
1587     character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
1588     Setting it to the maximum of \number#1}%
1589   \@tempcntb=#1\relax
1590 }

```

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

```

\MT@ex@factor@ 1591 \def\MT@get@ex@opt{%
\MT@stretch@ 1592   \MT@set@listname
1593   \MT@ifdefined@n@TF{MT@ex@c@\MT@ex@c@name @factor}{%
\MT@shrink@ 1594     \MT@let@cn\MT@ex@factor@\MT@ex@c@\MT@ex@c@name @factor}%
\MT@step@ 1595     \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor/1000}%
\MT@auto@ 1596   }{%
1597     \let\MT@ex@factor@\MT@ex@factor
1598   }%
1599   \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
1600   \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
1601   \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
1602   \LaTeX-def \MT@requires@luatex3\relax{%
1603     \MT@get@ex@opt@{auto}{\MT@ifstreql\MT@auto@}{autoexpand}{En}{Dis}abling automatic expansion}%
1604   \LaTeX-def }%
1605   \MT@ifdefined@n@T{MT@ex@c@\MT@ex@c@name @preset}{%
1606     \MT@preset@ex
1607     \let\MT@reset@ef@codes\relax
1608   }%
1609 }

```

\MT@get@ex@opt@

```

1610 \def\MT@get@ex@opt@#1#2{%
1611   \MT@ifdefined@n@TF{MT@ex@c@\MT@ex@c@name @#1}{%
1612     \MT@let@nn{MT@#1@}{MT@ex@c@\MT@ex@c@name @#1}%
1613     \MT@vinfo{... : #2}%
1614   }{%
1615     \MT@let@nn{MT@#1@}{MT@#1}%
1616   }%
1617 }

```

\MT@set@ex@heirs

```

1618 \def\MT@set@ex@heirs#1{%
1619   \efcode\MT@font#1=\efcode\MT@font\MT@char
1620   \LaTeX-debug\MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
1621   \LaTeX-debug\MT@dinfo@n1{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
1622 }

```

\MT@preset@ex

```

1623 \def\MT@preset@ex{%
1624   \@tempcntb=\csname MT@ex@c@\MT@ex@c@name @preset\endcsname\relax
1625   \MT@scale@factor
1626   \MT@set@all@ex\@tempcntb
1627 }
1628 \LaTeX-def|pdfTeX-def|luatex-def

```

14.2.3 Interword spacing (glue)

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

```
1629 <*pdfTeX-def>
1630 \MT@requires@pdftex{
1631 \def\MT@spacing{\MT@maybe@do{sp}}
```

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

```
1632 \def\MT@set@sp@codes{%
1633 \MT@if@list@exists{%
1634 \MT@get@opt
1635 \MT@reset@sp@codes
1636 \MT@get@inh@list
1637 \MT@set@inputenc{c}%
1638 \MT@load@list\MT@sp@cc@name
1639 \MT@set@listname
1640 \MT@let@cn\@tempc\MT@sp@c@\MT@sp@c@name}%
1641 \expandafter\MT@set@codes\@tempc,\relax,%
1642 }\MT@reset@sp@codes
1643 }
```

`\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).

```
1644 \def\MT@sp@split@val#1,#2,#3\relax{%
1645 \def\@tempb{#1}%
1646 \MT@ifempty\@tempb\relax{%
1647 \MT@get@space@unit2%
1648 \MT@scale@to@em
1649 \knbscode\MT@font\MT@char=\@tempcntb
1650 <debug>\MT@info@n1{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
1651 }%
1652 \def\@tempb{#2}%
1653 \MT@ifempty\@tempb\relax{%
1654 \MT@get@space@unit3%
1655 \MT@scale@to@em
1656 \stbscode\MT@font\MT@char=\@tempcntb
1657 <debug>\MT@info@n1{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
1658 }%
1659 \def\@tempb{#3}%
1660 \MT@ifempty\@tempb\relax{%
1661 \MT@get@space@unit4%
1662 \MT@scale@to@em
1663 \shbscode\MT@font\MT@char=\@tempcntb
1664 <debug>\MT@info@n1{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
1665 }%
1666 \MT@ifdefined@c@T\MT@sp@inh@name{%
1667 \MT@ifdefined@nT\MT@inh@\MT@sp@inh@name @\MT@char @}%
1668 \MT@exp@cs\MT@map@tlist@c\MT@inh@\MT@sp@inh@name @\MT@char @\MT@set@sp@heirs
1669 }%
1670 }%
1671 }
```

`\MT@set@sp@heirs`

```
1672 \def\MT@set@sp@heirs#1{%
1673 \knbscode\MT@font#1=\knbscode\MT@font\MT@char
1674 \stbscode\MT@font#1=\stbscode\MT@font\MT@char
1675 \shbscode\MT@font#1=\shbscode\MT@font\MT@char
1676 <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
1677 <debug>\MT@info@n1{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
1678 <debug> \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
1679 }
```

`\MT@set@all@sp`

`\MT@reset@sp@codes` 1680 \def\MT@set@all@sp#1#2#3{%

`\MT@reset@sp@codes@`


```

1681 (debug)\MT@info@n1{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
1682 \let\MT@temp@empty
1683 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
1684 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
1685 \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
1686 \MT@do@font\MT@temp
1687 }
1688 \def\MT@reset@sp@codes@{\MT@set@all@sp@z@z@z@}
1689 \let\MT@reset@sp@codes\relax

\MT@preset@sp
\MT@preset@sp@
1690 \def\MT@preset@sp{%
1691 \expandafter\expandafter\expandafter\MT@preset@sp@
1692 \csname MT@sp@code\MT@sp@code@name @preset\endcsname\@nil
1693 }
1694 \def\MT@preset@sp@#1,#2,#3\@nil{%
1695 \ifx\MT@sp@unit@\empty
1696 \MT@warn@preset@t@width{sp}%
1697 \MT@ifempty{#1}{\let\@tempa\empty}{\MT@preset@aux@factor{#1}\@tempa}%
1698 \MT@ifempty{#2}{\let\@tempc\empty}{\MT@preset@aux@factor{#2}\@tempc}%
1699 \MT@ifempty{#3}{\let\@tempb\empty}{\MT@preset@aux@factor{#3}\@tempb}%
1700 \else
1701 \MT@ifempty{#1}{\let\@tempa\empty}{\MT@preset@aux@space2{#1}\@tempa}%
1702 \MT@ifempty{#2}{\let\@tempc\empty}{\MT@preset@aux@space3{#2}\@tempc}%
1703 \MT@ifempty{#3}{\let\@tempb\empty}{\MT@preset@aux@space4{#3}\@tempb}%
1704 \fi
1705 \MT@set@all@sp@\@tempa\@tempc\@tempb
1706 }
1707 }\relax

```

14.2.4 Additional kerning

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

```

1708 \MT@requires@pdftex6{
1709 \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

1710 \def\MT@set@kn@codes{%
1711 \MT@if@list@exists{%
1712 \MT@get@opt
1713 \MT@reset@kn@codes
1714 \MT@get@inh@list
1715 \MT@set@inputenc{c}%
1716 \MT@load@list\MT@kn@code@name
1717 \MT@set@listname
1718 \MT@let@cn\@tempc{\MT@kn@code\MT@kn@code@name}%
1719 \expandafter\MT@set@codes\@tempc,\relax,%
1720 }\MT@reset@kn@codes
1721 }

```

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

```

1722 \def\MT@kn@split@val#1,#2\relax{%
1723 \def\@tempb{#1}%
1724 \MT@ifempty\@tempb\relax{%
1725 \MT@get@space@unit2%
1726 \MT@scale@to@em
1727 \knbcode\MT@font\MT@char=\@tempcntb
1728 (debug)\MT@info@n1{4}{;;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
1729 }%
1730 \def\@tempb{#2}%
1731 \MT@ifempty\@tempb\relax{%
1732 \MT@get@space@unit2%
1733 \MT@scale@to@em
1734 \knaccode\MT@font\MT@char=\@tempcntb

```

```

1735 (debug)\MT@dinfol{4}{;;; knac (\MT@char): \number\knaccode\MT@font\MT@char: [#2]}%
1736 }%
1737 \MT@ifdefined@cT\MT@kn@inh@name{%
1738   \MT@ifdefined@nT\MT@inh@MT@kn@inh@name @\MT@char @}{%
1739     \MT@exp@cs\MT@map@tlist@c\MT@inh@MT@kn@inh@name @\MT@char @}\MT@set@kn@heirs
1740   }%
1741 }%
1742 }

\MT@set@kn@heirs

1743 \def\MT@set@kn@heirs#1{%
1744   \knbccode\MT@font#1=\knbccode\MT@font\MT@char
1745   \knaccode\MT@font#1=\knaccode\MT@font\MT@char
1746   (debug)\MT@dinfol{2}{-- heir of \MT@char: #1}%
1747   (debug)\MT@dinfol{4}{;;; knbc (#1): \number\knbccode\MT@font\MT@char/%
1748   (debug)                               \number\knaccode\MT@font\MT@char}%
1749 }

\MT@set@all@kn

\MT@reset@kn@codes 1750 \def\MT@set@all@kn#1#2{%
\MT@reset@kn@codes@ 1751 (debug)\MT@dinfol{3}{-- knac/knbc: setting all to #1/#2}%
1752   \let\MT@temp@empty
1753   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbccode\MT@font\@tempcnta=#1\relax}}%
1754   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knaccode\MT@font\@tempcnta=#2\relax}}%
1755   \MT@do@font\MT@temp
1756 }
1757 \def\MT@reset@kn@codes@\MT@set@all@kn\z@\z@}
1758 \let\MT@reset@kn@codes\relax

\MT@preset@kn

\MT@preset@kn@ 1759 \def\MT@preset@kn{%
1760   \expandafter\expandafter\expandafter\MT@preset@kn@
1761   \csname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
1762 }
1763 \def\MT@preset@kn@#1,#2\@nil{%
1764   \ifx\MT@kn@unit@\@empty
1765     \MT@warn@preset@twidth{kn}%
1766     \let\MT@preset@aux\MT@preset@aux@factor
1767   \else
1768     \def\MT@preset@aux{\MT@preset@aux@space2}%
1769   \fi
1770   \MT@ifempty{#1}\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
1771   \MT@ifempty{#2}\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
1772   \MT@set@all@kn\@tempa\@tempb
1773 }
1774 }\relax
1775 (/pdfTeX-def)

```

14.2.5 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```

1776 (*pdfTeX-def|luatex-def)
1777 (pdfTeX-def)\MT@requires@pdfTeX6
1778 (luatex-def)\MT@requires@luatex3
1779 {

```

```

\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 1780 \let\MT@tr@font@list\@empty
1781 \def\MT@tracking@{%
1782   \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
1783   \ifMT@in@list\else
1784     \MT@maybe@do{tr}%

```

```

1785 \ifMT@do\else
1786 \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
1787 \fi
1788 \fi
1789 }
1790 </pdfTeX-def|LaTeX-def>
1791 <pdfTeX-def|LaTeX-def|letterspace>\let\MT@tracking
1792 <pdfTeX-def|LaTeX-def> \MT@tracking@
1793 <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.

Tracking won't work if the original font's `\fontdimen 6` is zero, in which case we issue a warning (once for every font).

```

1794 <*pdfTeX-def|LaTeX-def|letterspace>
1795 \def\MT@set@tr@codes{%
1796 <*pdfTeX-def|LaTeX-def>
1797 \MT@vinfo{Tracking font `~\MT@@font'\on@line}%
1798 \MT@ifdefined@n@TF{\MT@@font-fake6}{%
1799 \expandafter\ifx\csname\MT@@font-fake6\endcsname\@empty
1800 \MT@warning@n1{%
1801 Font `~\MT@@font' does not specify its\MessageBreak
1802 \@backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
1803 \@nameuse{MT@abbr@MT@feat} will not work with this font}%
1804 \MT@gl@et@nc{\MT@@font-fake6}\relax
1805 \fi
1806 }{%
1807 \MT@if@list@exists
1808 \MT@get@tr@opt
1809 \relax
1810 </pdfTeX-def|LaTeX-def>
1811 \MT@ifdefined@c@TF\MT@letterspace@relax{\let\MT@letterspace@MT@letterspace}%
1812 \ifnum\MT@letterspace@=\z@

```

Zero tracking requires special treatment.

```

1813 \MT@set@tr@zero
1814 \else
1815 <pdfTeX-def|LaTeX-def> \MT@vinfo{... Tracking by \number\MT@letterspace@}%

```

Letterspacing only works in PDF mode.

```

1816 \MT@warn@tracking@DVI

```

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

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

```

1817 \xdef\MT@lsfont{\csname\expandafter\string\font@name
1818 /\number\MT@letterspace@ls\endcsname}%
1819 \expandafter\ifx\MT@lsfont\relax
1820 <debug>\MT@dinfo@n1{1}{... new letterspacing instance}%

```

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

```

1821 \MT@get@ls@basefont

```

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

```

1822 <*LaTeX-def|letterspace>
1823 \MT@if@luaotf@font{%
1824 <LaTeX-def&debug>\MT@dinfo@n1{1}{... luaotf font: \MessageBreak

```

```

1825 <luatex-def&debug> \expandafter\fontname\font@name}%
1826 \ifnum\MT@letterspace@<z@\def\MT@minus{-}\else\let\MT@minus\empty\fi
1827 \global\expandafter\font\MT@lsfont=%
1828 \expandafter\MT@exp@two@c\expandafter\MT@ls@fontspec@font
1829 \expandafter\fontname\expandafter\font@name\space \@nil
1830 }{%
1831 </luatex-def|letterspace>
1832 <luatex-def&debug>\MT@dinfol{1}{... legacy font}%
1833 \global\expandafter\letterspacefont\MT@lsfont\font@name\MT@letterspace@
1834 <luatex-def|letterspace> }%

```

Scale interword spacing (not configurable in letterspace).

```

1835 <*pdfTEX-def|luatex-def>
1836 \MT@ifdefined@c@TF\MT@tr@ispace
1837 {\let\@tempa\MT@tr@ispace}%
1838 {\edef\@tempa{\MT@letterspace@*,,}}%
1839 \MT@ifdefined@c@TF\MT@tr@ospace
1840 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
1841 {\edef\@tempa{\@tempa,,}}%
1842 \expandafter\MT@tr@set@space\@tempa,%
1843 </pdfTEX-def|luatex-def>
1844 <*letterspace>
1845 % spacing = {<letterspace amount>*,,}
1846 \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@letterspace@\relax sp
1847 * \fontdimen2\MT@lsfont/1000\relax
1848 </letterspace>

```

Adjust outer kerning (microtype only).

```

1849 <*pdfTEX-def|luatex-def>
1850 \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
1851 \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

1852 \MT@ifdefined@c@T\MT@tr@ligatures\MT@tr@noligatures
1853 </pdfTEX-def|luatex-def>
1854 <*letterspace>
1855 % no ligatures = {f}
1856 \tagcode\MT@lsfont`f=\m@ne
1857 </letterspace>

```

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

```

1858 <luatex-def|letterspace> \MT@if@luaotf@font\relax{%
1859 <debug>\MT@dinfol{2}{... compensating for tracking (\number\MT@letterspace@)}%
1860 \MT@do@font{\lpcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax
1861 \rprcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax}%
1862 \let\MT@the@pr@code\MT@the@pr@code@tr
1863 <luatex-def|letterspace> }%
1864 \fi

```

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

```

1865 \aftergroup\MT@set@lsfont
1866 <pdfTEX-def|luatex-def> \let\MT@font\MT@lsfont
1867 <luatex-def> \MT@if@luaotf@font\MT@font\relax

```

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

```

\MT@curr@ls 1868 \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@letterspace@}}%
1869 \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.

```

1870 <*pdfTEX-def|luatex-def>
1871 \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax

```

```

1872 \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
1873 \MT@tr@outer@l
1874 </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'.

```

1875 \ifx\MT@ls@adjust\@empty
1876 <letterspace> % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
1877 \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\font@name/2000\relax
1878 \MT@ls@outer@k

```

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

```

1879 <*pdfTeX-def|luatex-def>
1880 \else
1881 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
1882 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1883 \ifdim\MT@outer@kern=\z@\else \MT@ls@outer@k \fi
1884 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
1885 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1886 </pdfTeX-def|luatex-def>
1887 <*letterspace>
1888 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
1889 \MT@afteraftergroup{%
1890 \MT@set@curr@ok
1891 \noexpand\MT@ls@outer@k
1892 }%
1893 </letterspace>
1894 \fi
1895 <*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).

```

1896 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%

```

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

```

1897 \MT@afteraftergroup{%
1898 \MT@set@curr@os
1899 \MT@set@curr@ok
1900 \noexpand\MT@tr@outer@r
1901 }%
1902 </pdfTeX-def|luatex-def>
1903 \fi
1904 <pdfTeX-def|luatex-def> }%
1905 }

```

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

```

1906 \def\MT@afteraftergroup#1{%
1907 <!letterspace> \MT@maybe@gobble@with@tikz{%
1908 \MT@ifdefined@n@TF{MT@aftergroup@number\currentgrouplevel}\relax{%
1909 \MT@exp@cs\xdef{MT@aftergroup@number\currentgrouplevel}%
1910 {\MT@exp@cs\MT@gl@et{MT@aftergroup@number\currentgrouplevel}\noexpand\@undefined#1}%
1911 \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
1912 {MT@aftergroup@number\currentgrouplevel}%
1913 }%
1914 <!letterspace> }%
1915 }
1916 </pdfTeX-def|luatex-def|letterspace>

```

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

the colon ourselves).

```

1917 <*luatex-def|letterspace>
1918 \def\MT@ls@fontspec@colon#1:#2:#3:#4\@nil{\ifx\#3\#1:#2\else#1:#2:#3\fi}
1919 \def\MT@ls@fontspec@font#1 #2\@nil{%
1920   "\MT@ls@fontspec@colon#1::\relax\@nil
1921   kernfactor=\MT@minus \ifnum\MT@letterspace@=1000 1\else 0.%
1922   \ifnum\MT@minus\MT@letterspace@<100 0\fi
1923   \ifnum\MT@minus\MT@letterspace@<10 0\fi
1924   \number\MT@minus\MT@letterspace@ \fi;"
1925   \ifx\#2\ at \f@size pt\else#2\fi\relax
1926 }
1927 </luatex-def|letterspace>

```

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

```

1928 <*pdfTeX-def|luatex-def>
1929 \def\MT@get@tr@opt{%
1930   \MT@set@listname
1931   \let\MT@tr@factor@\m

```

`\MT@tr@unit@` Different unit (for letterspace and/or (outer)spacing)?

```

1932 \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name @unit}{%
1933   \MT@let@cn\MT@tr@unit@{MT@tr@c@\MT@tr@c@name @unit}%
1934   \ifdim\MT@tr@unit@=1em
1935     \let\MT@tr@unit@\@undefined
1936   \else
1937     \MT@get@unit\MT@tr@unit@
1938   \fi
1939 }%
1940 \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name}{%
1941   \MT@let@cn\MT@letterspace{MT@tr@c@\MT@tr@c@name}%
1942   \MT@ifdefined@c@T\MT@tr@unit@{%
1943     \let\@tempb\MT@letterspace
1944     \MT@scale@to@em
1945     \edef\MT@letterspace{\number\@tempcntb}%
1946   }%
1947 }%

```

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

```

\MT@tr@ospace 1948 \MT@get@tr@opt@{spacing} {ispace}%
1949 \MT@get@tr@opt@{outerspacing}{ospace}%

```

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

```

1950 \MT@get@tr@opt@{outerkerning}{okern}%

```

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

```

1951 \MT@get@tr@opt@{noligatures} {ligatures}%
1952 }

```

`\MT@get@tr@opt@`

```

1953 \def\MT@get@tr@opt@#1#2{%
1954   \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name @#1}%
1955   {\MT@let@nn{MT@tr@#2}{MT@tr@c@\MT@tr@c@name @#1}}%
1956 }
1957 </pdfTeX-def|luatex-def>

```

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

```

1958 <*pdfTeX-def|luatex-def|letterspace>
1959 <plain>\MT@requires@1atex2{
1960 \def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}

```

`\lstyle` Disable the tests whether the font should be letterspaced, then trigger the setup. Only `\textls` can be used in math mode (`\lstyle` may be used inside another text switch, of course). Still, we have to ensure that math fonts are set up again.

Setting `\glb@currsiz` globally to `\@empty` (our previous solution) could throw us into an infinite loop (e.g., with the `psnfss` packages, via `\every@math@size`), so we issue `\glb@set tings` instead. However, in certain situations, we may still miss some math fonts, so let's try to also enforce it by emptying `\glb@currsiz`, fingers crossed. The overhead seems small.

```

1961 \DeclareRobustCommand\lssstyle{%
1962   \not@math@alphabet\lssstyle\textls
1963   \let\glb@currsiz\@empty
1964   \pdfTeX-def\luatex-def \MT@maybe@gobble@with@tikz{\aftergroup\glb@set tings}%
1965   \pdfTeX-def\luatex-def \def\MT@feat{tr}%
1966   \let\MT@tracking\MT@set@tr@codes
1967   \selectfont
1968 }

```

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

```

1969 (*plain)
1970 {}
1971 \def\MT@set@lfont{\MT@lfont}
1972 \def\lssstyle{%
1973   \begingroup
1974   \escapechar@m@ne
1975   \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
1976   \MT@set@tr@codes
1977   \endgroup
1978 }
1979 \let\textls\undefined
1980 \let\lslig\undefined
1981 }
1982 </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.

```

1983 \DeclareRobustCommand\lslig[1]{%
1984   {\MT@ifdefined@c@TF\MT@curr@lfont}%
1985   \escapechar@m@ne
1986   \MT@get@lfont@basefont
1987   \MT@outer@kern=\dimexpr\MT@curr@lfont sp * \fontdimen6\font@name/2000\relax
1988   \kern\MT@outer@kern
1989   \font@name #1%
1990   \kern\MT@outer@kern
1991   }{#1}}%
1992 }

```

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

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

```

1993 \def\MT@get@lfont@basefont{%
1994   \xdef\MT@lfont@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1995   \expandafter\ifx\MT@lfont@basefont\relax
1996   \MT@exp@two@c\MT@gl@let\MT@lfont@basefont\font@name
1997   \else
1998   \debug\MT@info@n1{... fixing base font}%
1999   \MT@exp@two@c\let\font@name\MT@lfont@basefont
2000   \fi
2001 }

```

`\MT@set@lfont@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
2002 \def\MT@set@lfont@basefont{\MT@exp@two@c\let\font@name\MT@lfont@basefont}
2003 \def\MT@set@tr@zero{%

```

```

2004 <debug>\MT@dinfoln1{... zero tracking}%
2005 \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2006 \expandafter\ifx\MT@ls@basefont\relax \else
2007 <debug>\MT@dinfoln1{... fixing base font}%
2008 \aftergroup\MT@set@lsbasefont
2009 \fi
2010 }
2011 </pdfTeX-def|luatex-def|letterspace>

```

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

```

2012 <pdfTeX-def|luatex-def>
2013 <pdfTeX-def>\MT@requires@pdfTeX7{
2014 \def\MT@tr@noligatures{%
2015 \ifx\MT@tr@ligatures\@empty
2016 \MT@noligatures@\MT@lsfont\@undefined
2017 \else
2018 \MT@noligatures@\MT@lsfont\MT@tr@ligatures
2019 \fi
2020 }
2021 <pdfTeX-def>
2022 }{
2023 \def\MT@tr@noligatures{%
2024 \MT@warning@n1{%
2025 Disabling selected ligatures is only possible since\MessageBreak
2026 pdfTeX 1.40.4. Disabling all ligatures instead}%
2027 \MT@glet\MT@tr@noligatures\relax
2028 }
2029 }
2030 </pdfTeX-def>

```

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

```
2031 \newskip\MT@outer@space
```

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

```

2032 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
2033 <debug>\MT@dinfoln2{... orig. space: \the\fontdimen2\MT@lsfont,
2034 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
2035 <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
2036 \let\MT@temp\@empty
2037 \MT@tr@set@space@{#1}{#4}{2}\@empty
2038 \MT@tr@set@space@{#2}{#5}{3}\@plus
2039 \MT@tr@set@space@{#3}{#6}{4}\@minus
2040 \MT@glet\nc{\MT@outer@space\expandafter\string\font@name}\MT@temp
2041 <debug>\MT@dinfoln2{... inner space: \the\fontdimen2\MT@lsfont,
2042 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
2043 <debug>\MT@dinfoln2{... outer space: \MT@temp}%
2044 }

```

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

```

2045 \def\MT@tr@set@space@#1#2#3#4{%
2046 \MT@ifempty{#2}{%
2047 \MT@ifempty{#1}\relax{%
2048 \MT@tr@set@space@{#1}{#3}{1000}%
2049 \fontdimen#3\MT@lsfont=\@tempdima
2050 }%
2051 \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
2052 }{%
2053 \MT@tr@set@space@{#2}{#3}{2000}%
2054 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
2055 \MT@ifempty{#1}\relax{%
2056 \MT@tr@set@space@{#1}{#3}{1000}%

```



```

2057 \fontdimen#3\MT@lsfont=\@tempdima
2058 }%
2059 }%
2060 }

```

`\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.

```

2061 \def\MT@tr@set@space@@#1#2#3{%
2062 \MT@test@ast#1*\@nil{%
2063 \MT@ifdefined@c@TF\MT@tr@unit@
2064 {\edef\@tempb{#1}\MT@scale@to@em}
2065 {\@tempcntb=#1\relax}%
2066 \@tempdima=\dimexpr\@tempcntb sp*\MT@dimen@six/1000\relax

```

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

```

2067 \ifnum#2=\tw@
2068 \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
2069 \fi
2070 }{%
2071 \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
2072 \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
2073 }%
2074 <debug>\MT@dinfo@n13{... : font dimen #2 (#1): \the\@tempdima}%
2075 }

```

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

```

2076 \def\MT@tr@outer@1{%
2077 \ifhmode
2078 \ifdim\lastskip>5sp
2079 \edef\x{\the\lastskip minus 0pt}%
2080 \setbox\z@\hbox{\MT@outer@space=\x}%
2081 \ifdim\wd\z@>\z@
2082 <debug>\MT@dinfo2{[[[ adjusting pre space: \the\MT@outer@space}%
2083 \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

2084 \let\MT@ls@outer@k\relax
2085 \else

```

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

```

2086 \ifdim\lastskip=%
2087 \ifnum\spacefactor<2000
2088 \spaceskip
2089 \else
2090 \ifdim\xspaceskip=\z@
2091 \dimexpr\spaceskip+\fontdimen7\font@name\relax
2092 \else
2093 \xspaceskip
2094 \fi
2095 \fi
2096 <debug>\MT@dinfo2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
2097 \unskip \hskip\MT@outer@space\relax
2098 \let\MT@ls@outer@k\relax
2099 \fi
2100 \fi
2101 \fi
2102 \fi
2103 }

```

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

`\MT@tr@outer@r`

(e.g., `\textsc`).

```
2104 \def\MT@tr@outer@r{%
2105   \futurelet\MT@tr@outer@next\MT@tr@outer@r@
2106 }
```

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

```
2107 \def\MT@if@outer@next#1{%
2108   \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2109 }
```

`\MT@tr@outer@r@`

```
2110 \def\MT@tr@outer@r@{%
2111   \def\MT@temp*{}}%
```

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

```
2112 \ifmmode \else
```

A similar bug occurred when adjustment would happen inside a discretionary group, which we prevent here. This only works with e-TeX (which we know is available).

```
2113   \ifnum\currentgrouptype=10 \else
2114     \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
2115 (debug)\MT@dinfo2{}}] adjusting post space (1): \the\MT@outer@space}%
2116     \fi}%
2117     \expandafter\ifcat\expandafter\noexpand\cname \MT@tr@outer@next\endcsname\egroup
2118     \ifhmode\unkern\fi\egroup
2119     \MT@set@curr@ok \MT@set@curr@os
2120     \def\MT@temp*{\afterassignment\MT@tr@outer@r@let\MT@temp=}%
2121     \else
```

If the next token is `\maybe@ic` (from an enclosing text command), we gobble it, read the next one, feed it to `\maybe@ic@` (via `\MT@tr@outer@icr`) and then call ourselves again.

```
2122   \MT@if@outer@next\maybe@ic{%
2123     \MT@set@curr@ok \MT@set@curr@os
2124     \def\MT@temp*{\afterassignment\MT@tr@outer@icr\let\MT@temp=}%
2125     }%
```

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).

```
2126   \MT@if@outer@next\check@icr{%
2127     \def\MT@temp*{\aftergroup\MT@tr@outer@r@check@icr\let\MT@temp=}%
2128     }%
2129     \MT@if@outer@next\@sptoken{%
2130       \def\MT@temp* {\ifhmode\hskip\MT@outer@space
2131 (debug)\MT@dinfo2{}}] adjusting post space (2): \the\MT@outer@space}%
2132       \fi}%
2133     }%
2134     \MT@if@outer@next~{%
2135       \def\MT@temp*~{\nobreak\hskip\MT@outer@space
2136 (debug)\MT@dinfo2{}}] adjusting post space (3): \the\MT@outer@space}%
2137       }%
2138     }%
2139     \MT@if@outer@next\relax{%
2140       \MT@if@outer@next\space\relax{%
2141         \MT@if@outer@next\xobeysp\relax{%
```

`xspace` requires special treatment.

```
2142   \MT@if@outer@next\xspace{%
```

```
2143             \def\MT@temp*\xspace{\MT@xspace}%
2144             }{%
```

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

```
2145             \def\MT@temp*{\ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k
2146 (debug)\MT@dinfo2{--- adjusting post kern: \the\MT@outer@kern}%
2147             \fi}%
2148             \MT@let@nc{\MT@tr@outer@next}\relax
2149             }}}}]]}}\fi
2150     \fi\fi
2151     \MT@temp*%
2152 }
```

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

```
\MT@tr@outer@icr@ 2153 \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@\MT@tr@outer@r}
2154 \def\MT@tr@outer@icr@{%
2155     \let@let@token= \MT@tr@outer@next
2156     \maybe@ic@
2157 }
```

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

```
2158 \def\MT@xspace{\futurelet\@let@token\MT@xspace@}
2159 \def\MT@xspace@{\@xspace@firsttrue\xspace
2160 \ifdim\lastskip>5sp
2161     \unskip \hskip\MT@outer@space
2162 \else
2163     \ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k \fi
2164 \fi
2165 }
```

For older pdf_TE_X versions and Lua_TE_X, throw an error.

```
2166 }{
2167     \DeclareRobustCommand\lsstyle{%
2168         \MT@error{Letterspacing only works with \MT@engine tex version
2169 (pdfTEX-def) 1.40%
2170 (luaTEX-def) 0.62%
2171         \MessageBreak or newer}
2172         {Upgrade \MT@engine tex, or try the `soul' package instead.}%
2173         \MT@glet\lsstyle\relax
2174     }
2175 }
```

And for X_Y_TE_X, too.

```
2176 (/pdfTEX-def|luaTEX-def)
2177 (*xetTEX-def)
2178 \DeclareRobustCommand\lsstyle{%
2179     \MT@error{Letterspacing currently doesn't work with xetex}
2180     {Run pdfTEX or luaTEX, or use the `soul' package instead.}%
2181     \MT@glet\lsstyle\relax
2182 }
2183 (/xetTEX-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.

```
\MT@ls@adjust@ 2184 (*package|letterspace)
2185 \DeclareRobustCommand\textls{%
2186     \ifstar{\let\MT@ls@adjust@\MT@ls@adjust@empty\MT@textls}%
2187     {\let\MT@ls@adjust@\MT@ls@adjust@relax\MT@textls}%
2188 }
```

`\MT@textls` This is now almost L_AT_EX's `\DeclareTextFontCommand`, with the difference that we
`\MT@letterspace@`

adjust the outer spacing and kerning also for `\lsstyle`, while L^AT_EX's text switches don't bother about italic correction.

```

2189 \newcommand\MT@textls[2] [] {%
2190   \ifmode
2191     \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
2192   \else
2193     \hmode@bgroup
2194       \MT@ls@set@ls{#1}%
2195       \lsstyle #2%
2196     \expandafter
2197     \egroup
2198   \fi
2199 }

```

`\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 2200 \def\MT@ls@adjust@empty{\let\MT@ls@adjust@empty}
\MT@ls@set@ls 2201 \def\MT@ls@adjust@relax{\let\MT@ls@adjust@relax}
2202 \def\MT@ls@set@ls#1{%
2203   \MT@ifempty{#1}%
2204   {\let\MT@letterspace@{\undefined}%
2205   {\KV@sp@def\MT@letterspace@{#1}%
2206   \edef\MT@letterspace@{\number\MT@letterspace@}%
2207   \MT@ls@too@large\MT@letterspace@}%
2208   \MT@ls@adjust@
2209 }

```

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

```

2210 \def\MT@ls@too@large#1{%
2211   \ifnum#1>\MT@tr@max
2212     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
2213     \let#1\MT@tr@max
2214   \else
2215     \ifnum#1<\MT@tr@min
2216       \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
2217       \let#1\MT@tr@min
2218     \fi
2219   \fi
2220 }

```

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

`\MT@tr@set@okern@`

```

2221 \newdimen\MT@outer@kern
2222 </package|letterspace>
2223 <*pdfTeX-def|luatex-def>
2224 \def\MT@tr@set@okern#1,#2,{%
2225   \let\MT@temp@empty
2226   \MT@ifempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
2227   \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
2228   \MT@gl@et@nc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
2229 <debug>\MT@dinfo@n12{... outer kerning: (#1,#2)}
2230 <debug> = \@nameuse{\MT@outer@kern\expandafter\string\font@name}}%
2231 }

```

`\MT@tr@set@okern@`

```

2232 \def\MT@tr@set@okern@#1{%
2233   \MT@test@ast#1*\@nil{%
2234     \MT@ifdefined@c@TF\MT@tr@unit@
2235     {\edef\@tempb{#1}\MT@scale@to@em}
2236     {\@tempcntb=#1\relax}%
2237     \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
2238   }%
2239   \MT@ifempty\@tempa{\let\@tempa\@m}\relax
2240   \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp

```

```

2241             * \fontdimen6\MT@lsfont/2000\relax
2242         }%
2243     \advance\@tempdima -\dimexpr \MT@letterspace@ sp
2244             * \fontdimen6\MT@lsfont/2000\relax
2245     \edef\MT@temp{\MT@temp{\the\@tempdima}}%
2246 }
2247 </pdfTeX-def|luatex-def>

```

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

```

2248 <(*pdfTeX-def|luatex-def|letterspace)>
2249 \def\MT@ls@outer@k{%
2250     \ifhmode
2251         \ifdim\lastkern=-3sp \unkern
2252         \ifdim\lastkern=3sp \kern-3sp
2253             \expandafter\expandafter\expandafter\@gobble
2254         \else \unkern
2255             \expandafter\expandafter\expandafter\@firstofone
2256         \fi
2257     \else
2258         \expandafter\@firstofone
2259     \fi
2260     {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
2261 \fi
2262 }
2263 </pdfTeX-def|luatex-def|letterspace>

```

14.2.6 Disabling ligatures

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

```

2264 <(*pdfTeX-def|luatex-def)>
2265 <pdfTeX-def>\MT@requires@pdfTeX5{
2266     \def\MT@noligatures{%
2267         \MT@dotrue
2268         \let\@tempa\MT@n1@setname
2269         \MT@map@clist@n{font,encoding,family,series,shape,size}{%
2270             \MT@ifdefined@nTF{MT@checklist@##1}%
2271                 {\csname MT@checklist@##1\endcsname}%
2272                 {\MT@checklist@{##1}}%
2273             {n1}}%
2274     }%
2275     \ifMT@do
2276         \MT@noligatures@\MT@font\MT@n1@ligatures
2277     \fi
2278 }

```

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

```

2279 <luatex-def>\MT@requires@luatex4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
2280 \def\MT@noligatures@#1#2{%
2281     \MT@ifdefined@c@TF#2{%

```

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

```

2282     \MT@ifdefined@c@TF\tagcode{%

```

No 'inputenc' key.

```

2283     \let\MT@warn@maybe@inputenc\empty
2284     \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
2285     \MT@map@clist@c#2{%
2286         \KV@sp@def\@tempa{##1}\MT@get@slot
2287         \ifnum\MT@char>\m@ne
2288             \tagcode#1\MT@char=\m@ne

```

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

```

2289 (\luatex-def)          \MT@if@luaotf@font
2290 (\luatex-def)          {\MT@lua{microtype.noligatures([[#1]],[[\MT@char]])}}\relax
2291          \fi
2292          }%
2293          \MT@vinfo{... Disabling ligatures for characters: #2}%
2294          }{%
2295          \pdfnoligatures#1%
2296          \MT@warning{Cannot disable selected ligatures (pdfTeX doesn't\MessageBreak
2297          know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
2298          the font instead}%
2299          }%
2300          }{%
2301          \pdfnoligatures#1%
2302 (\luatex-def)          \MT@if@luaotf@font
2303 (\luatex-def)          {\MT@lua{microtype.noligatures([[#1]],"_all_")}}\relax
2304          \MT@vinfo{... Disabling all ligatures}%
2305          }%
2306          }
2307 (\pdfTeX-def)}\relax
2308 (\pdfTeX-def|luatex-def)

```

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

```

2309 (*luafile)
2310 microtype.ligs = microtype.ligs or { }
2311
2312 local function noligatures(fontcs,liga)
2313   local fontcs = match(fontcs,"([^\ ]+)")
2314   microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
2315   table.insert(microtype.ligs[fontcs],liga)
2316 end
2317 microtype.noligatures = noligatures
2318
2319 local function keepligature(c)
2320   local nodedirect = node.direct
2321   local getfield   = nodedirect.getfield
2322   local getfont    = nodedirect.getfont
2323   local f,ch
2324   if type(c) == "userdata" then -- in older luaotfload versions, c was a node
2325     f = c.font
2326     ch = c.components.char
2327   else -- since 2.6, c is a (direct node) number
2328     f = getfont(c)
2329     ch = getfield(getfield(c,"components"),"char")
2330   end
2331   -- if ch then -- should always be true
2332   local ligs = microtype.ligs[match(tex.fontidentifier(f),"\\([^\ ]+)")]
2333   if ligs then
2334     for _,lig in pairs(ligs) do
2335       if lig == "_all_" or tonumber(lig) == ch then
2336         return false
2337       end
2338     end
2339   end
2340   return true
2341 -- end
2342 end
2343
2344 if luaotfload and luaotfload.letterspace then
2345   if luaotfload.letterspace.keepligature then

```

```

2346     microtype.info("overwriting function `keepligature'")
2347 end
2348 luaotfload.letterspace.keepligature = keepligature
2349 end
2350
2351 /luafile

```

14.2.7 Loading the configuration

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

```

2352 (*package)
2353 \def\MT@load@list#1{%
2354   \edef\@tempa{#1}%
2355   \MT@let@cn\@tempb{MT@MT@feat @c@\@tempa @load}%
2356   \MT@ifstreq\@tempa\@tempb{%
2357     \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\@tempa' cannot load itself}{}%
2358   }{%
2359     \ifx\@tempb\relax \else
2360       \MT@ifdefined@n@TF{MT@MT@feat @c@\@tempb}{%
2361         \MT@vinfo{... : First loading \@nameuse{MT@abbr@MT@feat} list `@\@tempb'}%
2362         \begin@group
2363           \MT@load@list\@tempb
2364         \end@group
2365         \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list
2366           \noexpand\MessageBreak`@\@tempb'}%
2367         \MT@let@cn\@tempc{MT@MT@feat @c@\@tempb}%
2368         \expand@after\MT@set@codes\@tempc,\relax,%
2369       }{%
2370         \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\@tempb' undefined.\MessageBreak
2371           Cannot load it from list `@\@tempa'}{}%
2372       }%
2373     \fi
2374   }%
2375 }

```

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

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

```

2376 \let\MT@file@list\@empty
2377 \def\MT@find@file#1{%
2378
2379   Check for existence of the file only once.

```

```

2378   \MT@in@clist{#1}\MT@file@list
2379   \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.

```

2380   \MT@begin@catcodes
2381   \let\MT@begin@catcodes\relax
2382   \let\MT@end@catcodes\relax
2383   \InputIfFileExists{mt-#1.cfg}{%
2384     \edef\MT@curr@file{mt-#1.cfg}%
2385     \MT@vinfo{... Loading configuration file \MT@curr@file}%
2386     \MT@xadd\MT@file@list{#1,%
2387   }{%
2388     \MT@get@basefamily#1\@empty\@empty\@empty\@nil
2389     \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
2390     \ifMT@inlist@
2391       \MT@xadd\MT@file@list{#1,%
2392     \else
2393       \InputIfFileExists{mt-\@tempa.cfg}{%
2394         \edef\MT@curr@file{mt-\@tempa.cfg}%
2395         \MT@vinfo{... Loading configuration file \MT@curr@file}%
2396       \MT@xadd\MT@file@list{\@tempa,#1,%

```

```

2397     }{%
2398     \MT@vinfo{... No configuration file mt-#1.cfg}%
2399     \MT@xadd\MT@file@list{#1,}%
2400     }%
2401     \fi
2402     }%
2403     \endgroup
2404     \fi
2405 }

```

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

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

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

```

2406 \def\MT@cfg@catcodes{%
2407   \makeatletter
2408   \catcode`\^7%
2409   \catcode`\ 9%
2410   \catcode`\^^I9%
2411   \catcode`\^^M9%
2412   \catcode`\z@
2413   \catcode`\{@ne
2414   \catcode`\}\tw@
2415   \catcode`\#6%
2416   \catcode`\%14%
2417   \MT@map@tlist@n
2418   {\!"#$%&'(\)*+,-./\/:;|<|=>?\[ ]_`{|}~}%
2419   \@makeother
2420 }

```

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

```

2421 \def\MT@begin@catcodes{%
2422   \begingroup
2423   \MT@cfg@catcodes
2424 }

```

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

```

2425 \let\MT@end@catcodes\endgroup

```

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

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

```

2426 \def\MT@get@basefamily#1#2#3#4\@nil{%
2427   \ifx\@empty#4%
2428     \def\@tempa{#1#2#3}%
2429   \else
2430     \let\@tempa\@empty
2431     \edef\@tempb{#1#2#3#4}%
2432     \expandafter\MT@get@basefamily@\@tempb\@nil
2433   \fi
2434 }

```

`\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'.

Table 4:

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

```

2435 \def\MT@get@basefamily@#1#2\@nil{%
2436   \edef\@tempa{\@tempa#1}%
2437   \ifx\#2\expandafter\@gobble\else\expandafter\@firstofone\fi
2438   {\MT@in@tlist{#2}\MT@variants
2439    \ifMT@in@tlist\else\MT@get@basefamily@#2\@nil\fi}%
2440 }

```

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

```

\MT@get@listname@
2441 \def\MT@get@listname#1{%
2442   <debug>\MT@edinfo@n1{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
2443   \let\MT@listname\undefined
2444   \def\@tempb{#1}%
2445   \MT@map@tlist@c\MT@try@order\MT@get@listname@
2446 }
2447 \def\MT@get@listname@#1{%
2448   \expandafter\MT@next@listname#1%
2449   \ifx\MT@listname\undefined \else
2450     \expandafter\MT@tlist@break
2451   \fi
2452 }

```

`\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.

```

2453 \def\MT@try@order{%
2454   {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
2455   {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
2456 }

```

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

```

2457 \def\MT@next@listname#1#2#3#4{%
2458   \ifnum#1=\z@\MT@nofamilytrue\fi
2459   \edef\@tempa{\MT@encoding
2460   /\ifnum#1=\@ne \MT@family \fi
2461   /\ifnum#2=\@ne \MT@series \fi
2462   /\ifnum#3=\@ne \MT@shape \fi
2463   /\ifnum#4=\@ne *\fi
2464   \MT@context}%
2465   <debug>\MT@edinfo@n1{1}{trying \@tempa}%
2466   \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
2467     \MT@next@listname@#4%
2468   }%

```

Also try with an alias family.

```

2469   \ifnum#1=\@ne
2470     \ifx\MT@familyalias\empty \else
2471       \edef\@tempa{\MT@encoding
2472       /\MT@familyalias
2473       /\ifnum#2=\@ne \MT@series\fi
2474       /\ifnum#3=\@ne \MT@shape\fi
2475       /\ifnum#4=\@ne *\fi

```

```

2476             \MT@context}%
2477 (debug)\MT@edinfo{n1}{(alias) \@tempa}%
2478             \MT@ifdefined@nT{MT@\@tempb @\@tempa}{%
2479             \MT@next@listname@#4%
2480             }%
2481             \fi
2482             \fi
2483             }%
2484 }

```

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

```

2485 \def\MT@next@listname#1{%
2486   \ifnum#1=\@ne
2487     \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @size}%
2488     \ifMT@inlist@
2489       \let\MT@listname\MT@size@name
2490       \fi
2491     \else
2492       \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
2493       \fi
2494 }

```

`\MT@if@list@exists`

```

\MT@context 2495 \def\MT@if@list@exists{%
2496   \MT@let@cn\MT@context{MT@\MT@feat @context}%
2497   \MT@ifstreq{@}\MT@context{\let\MT@context\@empty}\relax
2498   \MT@get@listname{\MT@feat @c}%
2499   \MT@ifdefined@c@TF\MT@listname{%
2500     \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
2501     \ifMT@nonselected
2502       \MT@vinfo{... Applying non-selected expansion (list `~\MT@listname')}%
2503     \else
2504       \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list `~\MT@listname'}%
2505     \fi
2506     \@firstoftwo
2507   }%

```

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

```
2508   \MT@let@nc{MT@\MT@feat @c@name}\@empty
```

Don't warn if `selected=false`.

```

2509   \ifMT@nonselected
2510     \MT@vinfo{... Applying non-selected expansion (no list)}%
2511   \else

```

Tracking doesn't require a list, either.

```

2512   \MT@ifstreq\MT@feat{tr}\relax{%
2513     \MT@warning{I cannot find a \@nameuse{MT@abbr@\MT@feat} list
2514     for font\MessageBreak`~\MT@font'%
2515     \ifx\MT@context\@empty\else\space(context: `~\MT@context')\fi.
2516     Switching off\MessageBreak\@nameuse{MT@abbr@\MT@feat} for this font}%
2517   }%
2518   \fi
2519   \@secondoftwo
2520 }%
2521 }

```

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

```

\MT@context 2522 \def\MT@get@inh@list{%
2523   \let\MT@context\@empty
2524   \MT@get@listname{\MT@feat @inh}%
2525   \MT@ifdefined@c@TF\MT@listname{%
2526     \MT@edef@n{MT@\MT@feat @inh@name}{\MT@listname}%

```

```

2527 <debug>\MT@dinfol{1}{... Using \nameuse{MT@abbr@MT@feat} inheritance list
2528 <debug>         ^\MT@listname'}%
2529     \MT@let@cn\@tempc{MT@MT@feat @inh@MT@listname}%

```

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

```

2530     \ifx\@tempc\@empty \else
2531 <debug>\MT@dinfol{1}{parsing inheritance list ...}%

```

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

```

2532     \begingroup
2533     \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak^\MT@listname'}%
2534     \MT@set@inputenc{inh}%
2535     \expandafter\MT@inh@do\@tempc,\relax,%
2536     \MT@gl@et@nc{MT@MT@feat @inh@MT@listname}\@empty
2537     \endgroup
2538     \fi
2539 }%
2540 \MT@let@nc{MT@MT@feat @inh@name}\@undefined
2541 }%
2542 }

```

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@ 2543 \def\MT@get@slot{%
2544     \escapechar^^\
2545     \let\MT@char@\m@ne
2546     \MT@noesttrue

```

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

```

2547     \MT@toks=\expandafter{\@tempa}%

```

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

```

2548     \MT@exp@two@c\MT@is@active\string\@tempa\@nil

```

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

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

```

2549     \expandafter\MT@is@letter\@tempa\relax\relax
2550     \ifnum\MT@char@ < \z@

```

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

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

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

```

2551     \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
2552     \MT@is@symbol

```

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

```
2553     {\expandafter\MT@is@composite\@tempa\relax\relax}%
2554     \ifnum\MT@char@ < \z@
```

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

```
2555     \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2556     \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
2557     \fi
2558     \fi

2559     \let\MT@char\MT@char@
2560     \MT@get@slot@
2561     \escapechar\m@ne
2562 }
2563 </package>
```

`\MT@get@slot@`

```
2564 <*pdftex-def|luatex-def|xetex-def>
2565 \def\MT@get@slot@{%
```

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

```
2566 <xetex-def> \ifnum\XeTeXfonttype\MT@font=\z@
2567 \ifnum\MT@char > \m@ne
```

In Lua \TeX , it may also be a glyph name, prefixed with `'`.

```
2568 <*luatex-def>
2569     \ifnum\MT@char=47\relax
2570     \ifMT@norest \else
2571     \@tempcnta=\MT@lua{
2572     local glyph = microtype.name_to_slot([[ \expandafter\@gobble\@tempa ]],true)
2573     if glyph then tex.write(glyph)
2574     else tex.write(-1)
2575     end
2576     }\relax
2577     \ifnum\@tempcnta<\z@
2578     \MT@warn@unknown
2579     \let\MT@char\m@ne
2580     \else
2581     \edef\MT@char{\the\@tempcnta}%
2582 <debug>\MT@dinfon1{3}{> \the\MT@toks' is a glyph name (\the\@tempcnta)}%
2583     \fi
2584     \fi
2585     \else
2586 </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.

```
2587     \ifMT@norest \else
2588     \MT@warn@rest
2589 <pdftex-def|luatex-def> \let\MT@char\m@ne
2590 <xetex-def> \let\MT@char\@empty
2591     \fi
2592 <luatex-def> \fi
2593     \else
2594     \MT@warn@unknown
2595 <xetex-def> \let\MT@char\@empty
2596     \fi
2597 <*xetex-def>
2598     \else
```

There are more possibilities for $X_{\text{}}\TeX$: It may also be a glyph name (prefixed

with ‘/’). We indicate this to `\MT@get@charwd` by reversing the sign of `\MT@char@`.

```

2599 \ifnum\MT@char=47\relax
2600 \ifMT@noreset \edef\MT@char{U47}%
2601 \else
2602 \@tempcnta=\XeTeXglyphindex"\expandafter\@gobble\@tempa"\relax
2603 \ifnum\@tempcnta=\z@
2604 \MT@warn@unknown
2605 \let\MT@char\@empty
2606 \else
2607 \edef\MT@char{\@tempa\space}%
2608 \edef\MT@char@{-\the\@tempcnta}%
2609 (debug)\MT@dinfoln{3}{> "\the\MT@toks' is a glyph name (\the\@tempcnta)}%
2610 \fi
2611 \fi
2612 \else
2613 \ifnum\MT@char > \m@ne
2614 \ifMT@noreset

```

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

```

2615 \@tempcnta=\XeTeXcharglyph\MT@char\relax
2616 \ifnum\@tempcnta=\z@
2617 \MT@info@missing@char
2618 \let\MT@char\@empty
2619 \else
2620 (debug)\MT@dinfoln{3}{> (glyph number: \the\@tempcnta,
2621 (debug) glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%
2622 \edef\MT@char{U\MT@char}%
2623 \fi
2624 \else
2625 \MT@warn@rest
2626 \let\MT@char\@empty
2627 \fi
2628 \else
2629 \MT@warn@unknown
2630 \let\MT@char\@empty
2631 \fi
2632 \fi
2633 \fi
2634 (/xetex-def)
2635 }
2636 (/pdftex-def|luatex-def|xetex-def)

```

This is the lua function to translate glyph name into slot number. Beginning with v2.2, `luaotfload` provides this function in an API, which we use if available, but (for now, at least) keep the old code for backward compatibility.

```

2637 (*luafile)
2638 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
2639 local slot_of_name = luaotfload.aux.slot_of_name
2640 microtype.name_to_slot = function(name, unsafe)
2641 return slot_of_name(font.current(), name, unsafe)
2642 end
2643 else
2644 -- we dig into internal structure (should be avoided)
2645 local function name_to_slot(name, unsafe)
2646 if fonts then
2647 local unicodes
2648 if fonts.ids then -- legacy luaotfload
2649 local tfmdata = fonts.ids[font.current()]
2650 if not tfmdata then return end
2651 unicodes = tfmdata.shared.otfdata.luatex.unicodes
2652 else -- new location
2653 local tfmdata = fonts.hashes.identifiers[font.current()]
2654 if not tfmdata then return end

```

```

2655     unicodes = tfmdata.resources.unicodes
2656     end
2657     local unicode = unicodes[name]
2658     if unicode then -- does the 'or' branch actually exist?
2659         return type(unicode) == "number" and unicode or unicode[1]
2660     end
2661     end
2662 end
2663 microtype.name_to_slot = name_to_slot
2664 end
2665
2666 (/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 2667 *(*pdf~~tex-def~~|lua~~tex-def~~|xet~~ex-def~~)*
2668 *\def\MT@max@char*
2669 *(pdf~~tex-def~~) {127 }*
2670 *(lua~~tex-def~~|xet~~ex-def~~) {1114111 }*
2671 *\def\MT@max@slot*
2672 *(pdf~~tex-def~~) {255 }*
2673 *(lua~~tex-def~~|xet~~ex-def~~) {1114111 }*
2674 *(/pdf~~tex-def~~|lua~~tex-def~~|xet~~ex-def~~)*

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

```

2675 (*package)
2676 \newif\ifMT@norest

2677 \def\MT@is@letter#1#2\relax{%
2678   \ifcat a\noexpand#1\relax
2679     \edef\MT@char@\number`#1%
2680     \ifx\#2\%
2681 (debug)\MT@info@n1{3}{> `the\MT@toks' is a letter (\MT@char@)}%
2682   \else
2683     \MT@norestfalse
2684   \fi
2685 \else
2686   \ifcat !\noexpand#1\relax
2687     \edef\MT@char@\number`#1%
2688 (debug)\MT@info@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
2689     \ifx\#2\%
2690       \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
2691     \else
2692       \MT@norestfalse
2693       \expandafter\MT@is@number#1#2\relax\relax
2694     \fi
2695   \fi
2696   \fi
2697 }

```

\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.

```

2698 \def\MT@is@number#1#2#3\relax{%
2699   \ifx\relax#3\relax \else
2700     \ifx\relax#2\relax \else
2701       \MT@noresttrue
2702       \if#1"\relax
2703         \def\x{\uppercase{\edef\MT@char@\number#1#2#3}}\x
2704 (debug)\MT@info@n1{3}{> ... a hexadecimal number: \MT@char@}%
2705       \else
2706         \if#1'\relax
2707           \def\MT@char@\number#1#2#3%

```

```

2708 (debug)\MT@info@n1{3}{> ... an octal number: \MT@char}%
2709     \else
2710         \MT@ifint{#1#2#3}{%
2711             \def\MT@char@{\number#1#2#3}%
2712 (debug)\MT@info@n1{3}{> ... a decimal number: \MT@char}%
2713         }\MT@noestfalse
2714     \fi
2715 \fi
2716 \ifnum\MT@char@ > \MT@max@slot
2717     \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
2718     \let\MT@char@\m@ne
2719 \fi
2720 \fi
2721 \fi
2722 }

```

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

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

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

```

2723 \def\MT@is@active#1#2@nil{%
2724     \ifnum\catcode`#1 = \active
2725         \begingroup
2726             \set@display@protect
2727             \let\IeC\@firstofone
2728             \let\@inpenc@undefined@\MT@undefined@char

```

Unicode handling has changed again with `LATEX 2019/10/01`.

```

2729     \let\UTF@two@octets@noexpand\@empty
2730     \let\UTF@three@octets@noexpand\@empty
2731     \let\UTF@four@octets@noexpand\@empty

```

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

```

2732     \def\UTFviii@defined##1{\ifx ##1\relax
2733         \MT@undefined@char{utf8}\else\expandafter ##1\fi}%

```

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

```

2734     \MT@ifdefined@c@T\PrerenderUnicode
2735         {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%

```

The `\expandafter` hocus-pocus should please `newunicodechar`.

```

2736     \edef\x{\endgroup
2737         \def\noexpand\@tempa{\expandafter\expandafter\expandafter\@empty\@tempa}%

```

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

```

2738         \MT@toks={\the\MT@toks\space(=
2739             \expandafter\expandafter\expandafter\@empty\@tempa)}%
2740     }%
2741     \x
2742 \fi
2743 }

```

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

```

2744 \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.

```
2745 \def\MT@is@symbol{%
2746   \expandafter\def\expandafter\MT@char\expandafter
2747     {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%
2748   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2749     \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
2750   \ifnum\MT@char@ < \z@
```

For TU encoding, the commands `\textquotesingle`, `\textasciigrave` and `\textquotedbl` are defined by means of the auxiliary macro `\remove@tlig`, which we take care of here.

```
2751   \expandafter\expandafter\expandafter\MT@is@tlig\MT@char\relax\relax
2752   \ifnum\MT@char@ < \z@
```

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

```
2753   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
2754   \fi
2755   \fi
2756 }
```

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

```
\MT@charstring 2757 \begingroup
2758   \catcode~\=\z@

2759   /MT@map@tlist@n{\CHARLEX}/@makeother
2760   /lowercase{%
2761     /def/x{/endgroup
2762     /def/MT@charstring{\CHAR"%
2763     /def/MT@is@char##1\CHAR"##2##3##4/relax{%
2764       /ifx/relax##4/relax
2765       /ifMT@xunicode
2766         /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
2767         /relax/relax/relax/relax/relax
2768       /fi
2769     /else
2770       /ifx/relax##1/relax
2771       /if##3\relax
2772       /edef/MT@char@{/number"##2}%
2773       /MT@ifstreq/MT@charstring{##3##4}/relax/MT@norestfalse
2774     /else
2775       /edef/MT@char@{/number"##2##3}%
2776       /MT@ifstreq/MT@charstring{##4}/relax
2777       {/MT@is@xchar##2##3|##4\CHAR"/relax}%
2778     /fi
2779 <debug> /MT@dinfo@n1{3}{>~/the/MT@toks' is a \char (/MT@char@)}%
2780     /fi
2781   /fi
2782   }%
```

`\MT@is@xchar` With `fontspec`'s TU encoding, glyph numbers may be up to four digits.

```
2783   /def/MT@is@xchar##1|##2\CHAR"##3##4/relax{%
2784     /MT@ifstreq/MT@charstring{##3##4}%
2785     {/edef/MT@char@{/number"##1##2}}/MT@norestfalse
2786   }%
```

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

```
\MT@strip@prefix 2787   /def/MT@charxstring{\CHAR "%
\MT@is@charx 2788   /def/MT@strip@prefix##1>##2/relax{##2}%
2789   /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
2790     /ifx/relax##1/relax
2791     /ifx/relax##6/relax/else
2792     /edef/MT@char@{/number"##2##3##4##5}%
```



```

2793           /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
2794 (debug)   /MT@dinfo@n1{3}{> `~/the/MT@toks' is a xunicode \char (/MT@char@)}%
2795           /fi
2796         /fi
2797       }%
2798     }%
2799   }
2800 /x

```

`\MT@is@tlig` This might have to change again with the next L^AT_EX release, ... or so I feared, but it still seems to be fine.

```

2801 \def\MT@is@tlig#1#2{%
2802   \ifx#1\remove@tlig
2803 (debug)   \MT@dinfo@n1{3}{> `~/the/MT@toks' (removing remove@tlig)}%
2804   \MT@is@number #2\relax\relax
2805   \fi
2806 }

```

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

```

2807 \def\MT@is@composite#1#2\relax{%
2808   \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 `\stringify`ing it. Thus, we will die gracefully even on wrong Unicode input without `utf8`.

```

2809   \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
2810     \string\csname\MT@encoding\endcsname
2811     \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%

```

In 2017, L^AT_EX introduced a new way of declaring accented Unicode commands (`\DeclareUnicodeComposite`), which we take care of here (`\UnicodeEncodingName` has been introduced at the same time):

```

2812   \ifx\UnicodeEncodingName\undefined\else
2813     \expandafter\expandafter\expandafter
2814     \MT@is@uni@comp\MT@char\iffontchar\else\fi\relax
2815   \fi
2816   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax

```

Again, xunicode.

```

2817   \ifnum\MT@char@ < \z@
2818     \ifMT@xunicode
2819       \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
2820       \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
2821       \MT@char\MT@charxstring\relax\relax\relax\relax\relax
2822     \fi
2823   \fi
2824   \fi
2825 }

```

`\MT@is@uni@comp` Helper for `\DeclareUnicodeComposite`.

```

2826 \def\MT@is@uni@comp#1\iffontchar#2\else#3\fi\relax{%
2827   \ifx\#2\\\else\edef\MT@char{\iffontchar#2\fi}\fi
2828 }

```

[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 `\mathchar`def'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 2829 \def\MT@set@list@name{%
2830   \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list\noexpand\MessageBreak
2831     \@nameuse{MT@MT@feat @c@name}}%
2832 }

```

`\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.

```

2833 \def\MT@warn@ascii{%
2834   \MT@warning@nl{Character `the\MT@toks' (= \MT@char@)
2835     is outside of ASCII range.\MessageBreak
2836     You must load the `inputenc' package before using\MessageBreak
2837     8-bit characters in \MT@curr@list@name}%
2838 }

```

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

```

2839 \def\MT@warn@number@too@large#1{%
2840   \MT@warning@nl{%
2841     Number #1 in encoding `MT@encoding' too large!\MessageBreak
2842     Ignoring it in \MT@curr@list@name}%
2843 }

```

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

```

2844 \def\MT@warn@rest{%
2845   \MT@warning@nl{%
2846     Unknown slot number of character\MessageBreak`the\MT@toks'%
2847     \MT@warn@maybe@inputenc\MessageBreak
2848     in font encoding `MT@encoding'.\MessageBreak
2849     Make sure it's a single character\MessageBreak
2850     (or a number) in \MT@curr@list@name}%
2851 }

```

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

```

2852 \def\MT@warn@unknown{%
2853   \MT@warning@nl{%
2854     Unknown slot number of character\MessageBreak`the\MT@toks'%
2855     \MT@warn@maybe@inputenc\MessageBreak
2856     in font encoding `MT@encoding' in \MT@curr@list@name}%
2857 }

```

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

```

2858 \def\MT@warn@maybe@inputenc{%
2859   \MT@ifdefined@n@T
2860   {MT@MT@feat @MT@cat @\csname MT@MT@feat @MT@cat @name\endcsname @inputenc}%
2861   { (input encoding \@nameuse
2862     {MT@MT@feat @MT@cat @\csname MT@MT@feat @MT@cat @name\endcsname @inputenc}')}%
2863 }

```

14.2.9 Hook into L^AT_EX's font selection

We append `\MT@setupfont` to `\pickup@font`, which is called by L^AT_EX every time a font is selected. We then check whether we've already seen this font, and if not, set it up for micro-typography. This ensures that we will catch all fonts, and that we will not set up fonts more than once. The whole package really hangs on this command.

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

For my reference:

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

Up to version 1.3 of this package, we were using `\define@newfont` as the hook, which is only called for *new* fonts, and therefore seemed the natural choice. However, this meant that we had to take special care to catch all fonts: we additionally had to set up the default font, the error font (if it wasn't the default font), we had to check for some packages that might have been loaded before `microtype` and were loading fonts, e.g., `jurabib`, `ledmac`, `pi font` (loaded by `hyperref`), `tipa`, and probably many more. Furthermore, we had to include a hack for the `IEEEtran` class which loads all fonts in the class file itself (to fine tune inter-word spacing), 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 2864 \let\MT@font@list\@empty
          2865 \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`.

```
2866 </package>
2867 <*package|letterspace>
2868 <plain>\MT@requires@latex2{
2869 \MT@addto@setup%
```

`\MT@orig@pickupfont` The `luatexja` package redefines `\char`, which will upset our parsing of text symbols and commands; instead of fixing this, we won't bother, at least for the moment, but simply issue a warning and disable all further warnings. The fix is left to the user by not specifying any text commands but only (Unicode) letters. The `xeCJK` package, or rather its `xunicode-addon`, also modifies the way text symbols are defined (like `luatexja` but in a different way). Again, we only issue a warning.

```
2870 <package> \MT@with@packageT{luatexja}{\MT@warn@unknown@once{luatexja}}%
```

```
2871 (package) \MT@with@package@T{xeCJK} {\MT@warn@unknown@once{xeCJK}}%
```

microtype also works with CJK in the sense that nothing will break when both packages are used at the same time. However, since CJK has its own way of encoding, it is currently not possible to create character-specific settings. That is, the only feature available with CJK fonts is (non-selected) expansion. (Tracking doesn't really work for other reasons.) Like us, CJK redefines `\pickup@font`.

```
2872 \ifpackage@loaded{CJK}{%
```

The `xeCJK` package in turn pretends that CJK was loaded, but does not change the definition of `\pickup@font`. With `xeCJK`, protrusion should be possible also for C/J/K characters; I haven't tried it, though.

```
2873 \ifpackage@loaded{xeCJK}{\@firstofone}{%
2874 \ifpackage@later{CJK}{2006/10/17}% 4.7.0
2875 {\def\MT@orig@pickupfont{\CJK@ifundefined\CJK@plane}}%
2876 {\def\MT@orig@pickupfont{\ifundefined{CJK@plane}}}%
2877 \g@addto@macro\MT@orig@pickupfont
2878 {\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).

```
2879 \ifpackage@loaded{CJKutf8}%
2880 {\ifpackage@later{CJKutf8}{2008/05/22}% 4.8.0
2881 {\ifpdf\expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}%
2882 {\@firstoftwo}}%
2883 {\@firstoftwo}%
2884 \g@addto@macro\MT@orig@pickupfont{%
2885 \expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2886 \define@newfont\else\xdef\font@name{%
2887 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}%
2888 \g@addto@macro\MT@orig@pickupfont{%
2889 \expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2890 \define@newfont\def\CJK@temp{v}%
2891 \ifx\CJK@temp\CJK@plane
2892 \expandafter\ifx\csname CJK@cmapp/\f@family\CJK@plane\endcsname\relax
2893 \else\csname CJK@cmapp/\f@family\CJK@plane\endcsname\fi
2894 \else\CJK@addcmapp\CJK@plane \fi
2895 \else\xdef\font@name{%
2896 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}%
2897 \@gobble
2898 }%
2899 }{\@firstofone}%
```

This is the normal L^AT_EX definition.

```
2900 {\def\MT@orig@pickupfont{\expandafter\ifx\font@name\relax\define@newfont\fi}}%
```

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

```
2901 \ifx\pickup@font\MT@orig@pickupfont \else
2902 \MT@warning@nl{%
2903 Command \string\pickup@font\space is not defined as expected.%
2904 \MessageBreak Patching it anyway. Some things may break%
2905 (*package)
2906 .\MessageBreak Double-check whether micro-typography is indeed%
2907 \MessageBreak applied to the document.%
2908 \MessageBreak (Hint: Turn on `verbose' mode)%
2909 (/package)
2910 }%
2911 \fi
```

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

```
2912 \g@addto@macro\pickup@font{\begingroup}%
```

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

```

2913 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
2914 \g@addto@macro\pickup@font{%
2915   \escapechar\m@ne
2916 }*package)
2917 <debug> \global\MT@inannottrue
2918 <debug> \MT@glet\MT@pdf@annot\@empty
2919 <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.

```

2920 \MT@let@cn\MT@font{MT@subst@\expandafter\string\font@name}%
2921 \ifx\MT@font\relax
2922   \let\MT@font\font@name
2923   \else
2924     \ifx\MT@font\font@name \else
2925 <debug> \MT@addto@annot{= substituted with \MT@font}%
2926     \MT@register@subst@font
2927     \fi
2928   \fi
2929   \MT@setupfont
2930 }/package)
2931 <letterspace> \MT@tracking
2932 \endgroup
2933 }%
2934 }*package)

```

`\MT@pickupfont` Remember the patched command, because we may have to disable ourselves in certain situations.

```

\MT@MT@pickupfont
\MT@ltx@pickupfont 2935 \let\MT@pickupfont\pickup@font
2936 \def\MT@MT@pickupfont {\let\pickup@font\MT@pickupfont}%
2937 \def\MT@ltx@pickupfont{\let\pickup@font\MT@orig@pickupfont}%

```

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

```

2938 \g@addto@macro\do@subst@correction
2939   {\edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
2940    \MT@glet@nc{MT@subst@\expandafter\string\font@name}\MT@font}%

```

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

```

2941 \let\MT@orig@add@accent\add@accent
2942 \def\add@accent#1#2{%
2943   \MT@ltx@pickupfont
2944   \MT@orig@add@accent{#1}{#2}%
2945   \MT@MT@pickupfont
2946 }%
2947 }/package)
2948 }
2949 <plain>\relax
2950 }*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.
2951 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}

\MT@register@font Register the current font.
2952 \def\MT@register@font{\xdef\MT@font@list{\MT@font@list\MT@font,}}

\MT@register@subst@font Register the substituted font (only if isn't registered already). Additionally, we
have to remove the substitute font from the list of fonts, so that we set it up again.
2953 \def\MT@register@subst@font{%
2954   \MT@exp@one@n\MT@in@clist\font@name\MT@font@list
2955   \ifMT@inlist@else
2956     \xdef\MT@font@list{\MT@font@list\font@name,}%
2957     \expandafter\MT@rem@from@clist\MT@font\MT@font@list
2958   \fi
2959 }

```

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.
2960 \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.
2961 \def\MT@check@font@cx{%
2962   \MT@if@true
2963   \MT@map@clist@c\MT@active@features{%
2964     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
2965     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2966     \ifMT@inlist@
2967       \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
2968     \else
2969       \MT@if@false
2970     \fi
2971   }%
2972   \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
2973 }

\MT@register@subst@font@cx Add the substituted font to each feature list and possibly remove substitute font.
2974 \def\MT@register@subst@font@cx{%
2975   \MT@map@clist@c\MT@active@features{%
2976     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
2977     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2978     \ifMT@inlist@ \else
2979       \MT@exp@cs\MT@xadd
2980       {MT@##1@\csname MT@##1@context\endcsname font@list}%
2981       {\font@name,}%
2982     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter\MT@font
2983     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2984     \fi
2985   }%
2986 }

\MT@register@font@cx For each feature, add the current font to the list, unless we didn't set it up.
2987 \def\MT@register@font@cx{%
2988   \MT@map@clist@c\MT@active@features{%
2989     \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
2990     \MT@exp@cs\MT@xadd

```

```

2991      {MT@##1@\csname MT@##1@context\endcsname font@list}%
2992      {\MT@font,}%
2993      \def\@tempa{##1}%
2994      \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
2995      \fi
2996      }%
2997      }

```

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

```

2998 \def\MT@maybe@rem@from@list#1{%
2999   \MT@ifstreq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
3000     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
3001     \MT@font \csname MT@\@tempa @#1font@list\endcsname
3002   }%
3003 }

```

`\microtypecontext` The user may change the context, so that different setups are possible. This is especially useful for multi-lingual documents.

`\MT@microtypecontext`

Inside the preamble, this command shouldn't actually do anything but remember itself for later.

```

3004 \def\microtypecontext{\MT@begin@catcodes\MT@microtypecontext}
3005 \def\MT@microtypecontext#1{\MT@end@catcodes\MT@addto@setup{\microtypecontext{#1}}}
3006 \MT@addto@setup{%
3007   \DeclareRobustCommand\microtypecontext{%
3008     \MT@begin@catcodes
3009     \MT@microtypecontext
3010   }%
3011   \def\MT@microtypecontext#1{%
3012     \MT@end@catcodes
3013     \MT@setup@contexts
3014     \let\MT@reset@context\relax

```

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

```

3015   \MT@glet\glsb@currsiz@empty
3016   \setkeys{MTC}{#1}%
3017   \selectfont
3018   \MT@reset@context
3019 }%
3020 }

```

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

```

\MT@textmicrotypecontext 3021 \DeclareRobustCommand\textmicrotypecontext{\MT@begin@catcodes\MT@textmicrotypecontext}
\MT@text@microtypecontext 3022 \def\MT@textmicrotypecontext#1{\MT@end@catcodes\MT@text@microtypecontext{#1}}
3023 \def\MT@text@microtypecontext#1#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@`

```

3024 \def\MT@reset@context@{%
3025   \MT@vinfo{<<< Resetting contexts\on@line
3026   <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
3027   <debug>           /\MT@tr@context/\MT@kn@context/\MT@sp@context
3028   }%
3029   \selectfont
3030 }

```

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

```

3031 \def\MT@setup@contexts{%
3032   \MT@map@clist@c\MT@active@features
3033   {\MT@glet@nc{MT@##1@font@list}\MT@font@list}%
3034   \MT@glet\MT@check@font\MT@check@font@cx
3035   \MT@glet\MT@register@font\MT@register@font@cx

```

```

3036 \MT@gl@et\MT@register@subst@font\MT@register@subst@font@cx
3037 \MT@gl@et\MT@setup@contexts\relax
3038 }

```

Define context keys.

```

3039 \MT@map@clist@c\MT@features@long{%
3040 \define@key{MTC}{#1}[]{}%
3041 \edef\@tempb{\@nameuse{MT@rbba@#1}}%
3042 \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
3043 \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).

```

3044 \MT@ifempty{#1}{\def\MT@val{0}}{\def\MT@val{#1}}%
3045 \MT@exp@cs\ifx{MT@\@tempb @context}\MT@val
3046 <debug>\MT@dinfo{1}{>>> no change of #1 context: `~\MT@val'}%
3047 \else
3048 \MT@vinfo{>>> Changing #1 context to `~\MT@val'\MessageBreak\on@line
3049 <debug> \space(previous: `~\@nameuse{MT@\@tempb @context}')}%
3050 }%
3051 \def\MT@reset@context{\aftergroup\MT@reset@context@}%

```

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

```

3052 \MT@gl@et@nn{MT@reset@\@tempb @codes}\MT@reset@\@tempb @codes@}%

```

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

```

3053 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
3054 \MT@val \cscname MT@\@tempb @doc@contexts\endcsname
3055 \ifMT@inlist@ \else
3056 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{\MT@val}}%
3057 <debug> \MT@dinfo{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
3058 \fi
3059 \MT@edef@n{MT@\@tempb @context}{\MT@val}%
3060 \fi
3061 \fi
3062 }%
3063 }

```

We also allow the activate shortcut.

```

3064 \define@key{MTC}{activate}[]{}%
3065 \setkeys{MTC}{protrusion={#1}}%
3066 \setkeys{MTC}{expansion={#1}}%
3067 }

```

\MT@pr@context Initialise the contexts.

\MT@ex@context 3068 \MT@exp@one@n\MT@map@clist@n{\MT@features,n1}}%

\MT@tr@context 3069 \MT@def@n{MT@#1@context}{@}%

\MT@sp@context 3070 \MT@def@n{MT@#1@doc@contexts}{@}%

\MT@kn@context 3071 }

\MT@pr@doc@contexts 3072 \let\MT@extra@context\@empty

\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.

```

3073 \def\DeclareMicrotypeSet{%

```



```

3074 \MT@begin@catcodes
3075 \ifstar
3076 \MT@DeclareSetAndUseIt
3077 \MT@DeclareSet
3078 }

\MT@DeclareSet
3079 \newcommand\MT@DeclareSet[3][]{%
3080 \MT@ifempty{#1}{%
3081 \MT@map@clist@c\MT@features{\MT@declare@sets{##1}{#2}{#3}}}%
3082 }{%
3083 \MT@map@clist@n{#1}{%
3084 \MT@ifempty{#1}\relax{%
3085 \MT@is@feature{##1}{set declaration `#2'}{%
3086 \MT@exp@one@n\MT@declare@sets
3087 {\csname MT@rbba@##1\endcsname}{#2}{#3}%
3088 }%
3089 }%
3090 }}%
3091 }%
3092 \MT@end@catcodes
3093 }

```

\MT@DeclareSetAndUseIt

```

3094 \newcommand\MT@DeclareSetAndUseIt[3][]{%
3095 \MT@DeclareSet[#1]{#2}{#3}%
3096 \UseMicrotypeSet[#1]{#2}%
3097 }

```

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

```
3098 \let\MT@curr@set@name\empty
```

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

```

3099 \def\MT@declare@sets#1#2#3{%
3100 \def\MT@curr@set@name{#2}%
3101 \MT@ifdefined@n@T{MT@#1@set@@\MT@curr@set@name}{%
3102 \MT@warning{Redefining \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
3103 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3104 \MT@glet@nc{MT@#1list@##1@\MT@curr@set@name}\@undefined
3105 }%
3106 }%
3107 \MT@glet@nc{MT@#1@set@@\MT@curr@set@name}\@empty
3108 (debug)\MT@dinfo{1}{declaring \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
3109 \setkeys{MT@#1@set}{#3}%
3110 }

```

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

```

3111 \def\MT@define@set@key@#1#2{%
3112 \define@key{MT@#2@set}{#1}[]{}%
3113 \MT@glet@nc{MT@#2list@#1@\MT@curr@set@name}\@empty
3114 \MT@map@clist@n{##1}{%
3115 \KV@sp@def\MT@val{###1}%
3116 \MT@get@highlevel{#1}%

```

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

```

3117 \MT@exp@two@n@g@addto@macro
3118 {\csname MT@#2list@#1@\MT@curr@set@name\expandafter\endcsname}%
3119 {\MT@val,}%
3120 }%

```

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

```

3121 \expandafter@g@addto@macro\expandafter\MT@font@sets
3122 \csname MT@#2list@#1@\MT@curr@set@name\endcsname
3123 (debug)\MT@dinfo@n1{1}{-- #1: \@nameuse{MT@#2list@#1@\MT@curr@set@name}}%
3124 }%

```

3125 }

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

3126 `\def\MT@get@highlevel#1{%`

3127 `\expandafter\MT@test@ast\MT@val*\@nil\relax{%`

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

3128 `\MT@ifempty\@tempa{\def\@tempa{#1}}\relax`

Test whether the command is actually defined.

3129 `\MT@ifdefined@n@TF{\@tempa default}%`

3130 `{\edef\MT@val{\expandafter\noexpand\csname \@tempa default\endcsname}}%`

3131 `{\MT@warning{\@backslashchar\@tempa default' is not a defined command.\MessageBreak`

3132 Ignoring `#1 = {\@tempa*}' in font set\MessageBreak``\MT@curr@set@name'}}`

3133 `\let\MT@val\@empty}%`

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

3134 `}%`

3135 }

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

3136 `\def\MT@test@ast#1*#2\@nil{%`

3137 `\def\@tempa{#1}%`

3138 `\MT@ifempty{#2}%`

3139 }

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

3140 `\let\MT@font@sets\@empty`

3141 `\def\MT@fix@font@set#1{%`

3142 `\MT@ifdefined@c@T{#1}{%`

3143 `\xdef#1{#1}%`

3144 `\ifMT@fontspec`

3145 `\xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%`

3146 `\fi`

3147 `\global\@onelevel@sanitize#1%`

3148 `}%`

3149 }

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

3150 `\def\MT@define@set@key@size#1{%`

3151 `\define@key{MT@#1@set}{size}[] {%`

3152 `\MT@map@clist@{##1}{%`

3153 `\def\MT@val{###1}%`

3154 `\expandafter\MT@get@range\MT@val--\@nil`

3155 `\ifx\MT@val\relax \else`

3156 `\MT@exp@cs\MT@xadd`

3157 `{\MT@#1list@size@\MT@curr@set@name}%`

3158 `{{{\MT@lower}{\MT@upper}\relax}}%`

3159 `\fi`

3160 `}%`

3161 `(debug)\MT@dinfo@n1{1}{-- size: \@nameuse{MT@#1list@size@\MT@curr@set@name}}%`

3162 `}%`

3163 }

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 does this for the OpenType version of Adobe’s Minion. (Available from CTAN at [pkg/minionpro](#)))

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

```

\MT@lower 3164 \def\MT@get@range#1-#2-#3\@nil{%
3165   \MT@ifempty{#1}{%
3166     \MT@ifempty{#2}{%
3167       \let\MT@val\relax
3168     }{%
3169       \def\MT@lower{0}%
3170       \def\MT@val{#2}%
3171       \MT@get@size
3172       \edef\MT@upper{\MT@val}%
3173     }%
3174   }{%
3175     \def\MT@val{#1}%
3176     \MT@get@size
3177     \ifx\MT@val\relax \else
3178       \edef\MT@lower{\MT@val}%
3179       \MT@ifempty{#2}{%
3180         \MT@ifempty{#3}%
3181         {\def\MT@upper{-1}}%

```

2048 pt is T_EX's maximum font size.

```

3182   {\def\MT@upper{2048}}%
3183 }{%
3184   \def\MT@val{#2}%
3185   \MT@get@size
3186   \ifx\MT@val\relax \else
3187     \MT@ifdim\MT@lower>\MT@val{%
3188       \MT@error{%
3189         Invalid size range (\MT@lower\space > \MT@val) in font set
3190         ~\MT@curr@set@name'.\MessageBreak Swapping sizes}}%
3191     \edef\MT@upper{\MT@lower}%
3192     \edef\MT@lower{\MT@val}%
3193   }{%
3194     \edef\MT@upper{\MT@val}%
3195   }%
3196   \MT@ifdim\MT@lower=\MT@upper
3197   {\def\MT@upper{-1}}%
3198   \relax
3199   \fi
3200 }%
3201 \fi
3202 }%
3203 }

```

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

```

3204 \def\MT@get@size{%
    A single star would mean \sizedefault, which doesn't exist, so we define it to be
    \normalsize.
3205   \if*\MT@val\relax
3206     \def\@tempa{\normalsize}%
3207   \else
3208     \MT@let@cn\@tempa{\MT@val}%
3209   \fi
3210   \ifx\@tempa\relax \else

```

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

```

3211   \begingroup
3212   \def\set@fontsize##1##2##3##4\@nil{\endgroup\def\MT@val{##2}}%

```

```
3213 \@tempa\@nil
3214 \fi
```

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

```
3215 \MT@ifdimen\MT@val{%
3216 \@defaultunits\@tempdima\MT@val pt\relax\@nnil
3217 \edef\MT@val{\strip@pt\@tempdima}%
3218 }{%
3219 \MT@warning{Could not parse font size `~\MT@val'\MessageBreak
3220 in font set `~\MT@curr@set@name'}%
3221 \let\MT@val\relax
3222 }%
3223 }
```

\MT@define@set@key@font

```
3224 \def\MT@define@set@key@font#1{%
3225 \define@key{MT@#1@set}{font}[]{%
3226 \MT@gl@et@nc{MT@#1@list@font\MT@curr@set@name}\@empty
3227 \MT@map@cl@ist@n{#1}{%
3228 \def\MT@val{###1}%
3229 \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*}}\relax
3230 \expandafter\MT@get@font\MT@val////\@nil
3231 \MT@exp@two@n@g@addto@macro
3232 {\csname MT@#1@list@font\MT@curr@set@name\expandafter\endcsname}%
3233 {\MT@val,}%
3234 }%
3235 \expandafter\g@addto@macro\expandafter\MT@font@sets
3236 \csname MT@#1@list@font\MT@curr@set@name\endcsname
3237 <debug>\MT@din@fo@n1{1}{-- font: \@nameuse{MT@#1@list@font\MT@curr@set@name}}%
3238 }%
3239 }
```

\MT@get@font Translate any asterisks.

```
3240 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
3241 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
3242 \ifx\MT@val\relax\def\MT@val{0}\fi
3243 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
3244 \let\MT@val\@tempb
3245 }
```

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

```
3246 \def\MT@get@font@#1#2#3#4#5#6{%
3247 \let\@tempb\@empty
3248 \def\MT@temp{#1/#2/#3/#4/#5}%
3249 \MT@get@axis{encoding}{#1}%
3250 \MT@get@axis{family}{#2}%
3251 \MT@get@axis{series}{#3}%
3252 \MT@get@axis{shape}{#4}%
3253 \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
3254 \MT@ifempty{#5}{%
3255 \MT@warn@axis@empty{size}{\string\normalsize}%
3256 \def\MT@val{*}%
3257 }{%
3258 \def\MT@val{#5}%
3259 }%
3260 \MT@get@size
3261 }
```

\MT@get@axis

```
3262 \def\MT@get@axis#1#2{%
3263 \def\MT@val{#2}%
3264 \MT@get@highlevel{#1}%
3265 \MT@ifempty\MT@val{%
```

```

3266 \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
3267 \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
3268 } \relax
3269 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
3270 }

```

\MT@warn@axis@empty

```

3271 \def\MT@warn@axis@empty#1#2{%
3272 \MT@warning{#1 axis is empty in font specification\MessageBreak
3273 ~\MT@temp'. Using `#2' instead}%
3274 }

```

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

```

3275 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
3276 \MT@define@set@key@{encoding}{#1}%
3277 \MT@define@set@key@{family}{#1}%
3278 \MT@define@set@key@{series}{#1}%
3279 \MT@define@set@key@{shape}{#1}%
3280 \MT@define@set@key@size{#1}%
3281 \MT@define@set@key@font{#1}%
3282 }

```

\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.

```

3283 \def\UseMicrotypeSet{%
3284 \MT@begin@catcodes
3285 \MT@UseMicrotypeSet
3286 }

```

\MT@UseMicrotypeSet

```

3287 \newcommand*\MT@UseMicrotypeSet[2][ ]{%
3288 \MT@ifempty{#1}{%
3289 \MT@map@clist@c\MT@features{\MT@use@set{##1}{#2}}}%
3290 }{%
3291 \MT@map@clist@n{#1}{%
3292 \MT@ifempty{##1}\relax{%
3293 \MT@is@feature{##1}{activation of set `#2'}{%
3294 \MT@exp@one@n\MT@use@set
3295 {\csname MT@rbba@##1\endcsname}{#2}%
3296 }%
3297 }%
3298 }%
3299 }%
3300 \MT@end@catcodes
3301 }

```

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

```

\MT@ex@setname 3302 \def\MT@use@set#1#2{%
\MT@tr@setname 3303 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
\MT@xdef@n{MT@#1@setname}{#2}%
\MT@sp@setname 3304 }{%
\MT@kn@setname 3305 }{%
\MT@kn@setname 3306 \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
\MT@use@set 3307 \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
3308 }%
3309 \MT@error{%
3310 The \@nameuse{MT@abbr@#1} set `#2' is undeclared.\MessageBreak
3311 Using set ~\@nameuse{MT@#1@setname}' instead}{%
3312 }%
3313 }

```

\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.

```

3314 \def\DeclareMicrotypeSetDefault{%

```

```

3315 \MT@begin@catcodes
3316 \MT@DeclareMicrotypeSetDefault
3317 }

```

\MT@DeclareMicrotypeSetDefault

```

3318 \newcommand*\MT@DeclareMicrotypeSetDefault[2] [] {%
3319 \MT@ifempty{#1}{%
3320 \MT@map@clist@ec\MT@features{\MT@set@default@set{##1}{#2}}{%
3321 }{%
3322 \MT@map@clist@en{#1}{%
3323 \MT@ifempty{##1}\relax{%
3324 \MT@is@feature{##1}{declaration of default set `#2'}{%
3325 \MT@exp@one@n\MT@set@default@set
3326 {\csname MT@rbb@##1\endcsname}{#2}%
3327 }%
3328 }%
3329 }}%
3330 }%
3331 \MT@end@catcodes
3332 }

```

\MT@default@pr@set

```

\MT@default@ex@set 3333 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 3334 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
\MT@default@sp@set 3335 (debug)\MT@dinfo{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}%
\MT@default@kn@set 3336 \MT@xdef@n{MT@default@#1@set}{#2}%
\MT@set@default@set 3337 }{%
3338 \MT@error{%
3339 The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak
3340 Cannot make it the default set. Using set\MessageBreak `all' instead}{}%
3341 \MT@xdef@n{MT@default@#1@set}{all}%
3342 }%
3343 }

```

14.3.2 Variants and aliases

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

\MT@variants

```

3344 \let\MT@variants\@empty
3345 \def\DeclareMicrotypeVariants{%
3346 \MT@begin@catcodes
3347 \ifstar
3348 \MT@DeclareVariants
3349 {\let\MT@variants\@empty\MT@DeclareVariants}%
3350 }

```

\MT@DeclareVariants

```

3351 \def\MT@DeclareVariants#1{%
3352 \MT@map@clist@en{#1}{%
3353 \def\@tempa{##1}%
3354 \@onelevel@sanitize\@tempa
3355 \xdef\MT@variants{\MT@variants{\@tempa}}%
3356 }%
3357 \MT@end@catcodes
3358 }

```

\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.

```

3359 \def\DeclareMicrotypeAlias{%
3360 \MT@begin@catcodes
3361 \MT@DeclareMicrotypeAlias
3362 }

```

`\MT@DeclareMicrotypeAlias`

```

3363 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
3364   \def\@tempb{#2}%
3365   \@onelevel@sanitize\@tempb
3366   \MT@ifdefined@n@T{MT@#1@alias}{%
3367     \MT@warning{Alias font family `\'@tempb' will override
3368       alias `\'@nameuse{MT@#1@alias}'\MessageBreak
3369       for font family `\'#1'}}%
3370   \MT@xdef@n{MT@#1@alias}{\@tempb}%

```

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

```

3371   \MT@ifdefined@c@T\MT@family{%
3372     <debug>\MT@dinfo{1}{Activating alias font `\'@tempb' for `\'@family'}%
3373     \MT@gl@et\MT@familyalias\@tempb
3374   }%
3375   \MT@end@catcodes
3376 }

```

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

```

3377 \def\LoadMicrotypeFile#1{%
3378   \edef\@tempa{\zap@space#1 \@empty}%
3379   \@onelevel@sanitize\@tempa
3380   \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3381   \ifMT@inlist@
3382     \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
3383   \else
3384     \MT@xadd\MT@file@list{\@tempa,}%
3385     \MT@begin@catcodes
3386     \InputIfFileExists{mt-\@tempa.cfg}{%
3387       \edef\MT@curr@file{mt-\@tempa.cfg}%
3388       \MT@vinfo{... Loading configuration file \MT@curr@file}%
3389     }{%
3390       \MT@warning{Configuration file mt-\@tempa.cfg\MessageBreak
3391         does not exist}%
3392     }%
3393     \MT@end@catcodes
3394   \fi
3395 }
3396 </package>
3397 </package|letterspace>

```

14.3.3 Disabling ligatures

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

`\MT@n1@setname` The optional argument may be used to disable selected ligatures only.

```

\MT@n1@ligatures 3398 <*pdfTeX-def|LaTeX-def>
3399 <pdfTeX-def>\MT@requires@pdfTeX5{
3400 \def\DisableLigatures{%
3401   \MT@begin@catcodes
3402   \MT@DisableLigatures
3403 }
3404 \newcommand*\MT@DisableLigatures[2][ ]{%
3405   \MT@ifempty{#1}\relax{\gdef\MT@n1@ligatures{#1}}%
3406   \xdef\MT@active@features{\MT@active@features,n1}%
3407   \global\MT@noLigaturestrue
3408   \MT@declare@sets{n1}{no ligatures}{#2}%
3409   \gdef\MT@n1@setname{no ligatures}%
3410   \MT@end@catcodes
3411 }

```

```

3412 <pdftex-def>}{
3413 </pdftex-def|luatex-def>

    If pdfTEX is too old, we throw an error.

3414 <*pdftex-def|xetex-def>
3415 \renewcommand*{DisableLigatures}[2][ ]{%
3416   \MT@error{Disabling ligatures of a font is only possible\MessageBreak
3417     with pdftex version 1.30 or newer.\MessageBreak
3418     Ignoring \string\DisableLigatures}{%
3419 <pdftex-def>   Upgrade
3420 <xetex-def>    Use
3421   pdftex.}%
3422 }
3423 <pdftex-def>}{
3424 </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.

```

3425 <*package>
3426 \def\DeclareMicrotypeBabelHook#1#2{%
3427   \MT@map@clist@n{#1}{%
3428     \KV@@sp@def\@tempa{##1}%
3429     \MT@gdef@n{MT@babel@\@tempa}{#2}%
3430   }%
3431 }
3432 </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).

```

3433 <*pdftex-def|xetex-def|luatex-def>
3434 \def\SetProtrusion{%
3435   \MT@begin@catcodes
3436   \MT@SetProtrusion
3437 }

```

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

```
\MT@pr@c@name 3438 \newcommand*\MT@SetProtrusion[3][ ]{%
```

```
\MT@extra@context 3439 \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.

```

3440   \MT@set@named@keys{MT@pr@c}{#1}%
3441 <debug>\MT@dinfo{1}{creating protrusion list `~\MT@pr@c@name'}%
3442   \def\MT@permutelist{pr@c}%
3443   \setkeys{MT@cfg}{#2}%

```

We have parsed the second argument, and can now define macros for all permutations of the font attributes to point to `\MT@pr@c@<name>`, ...

```
3444   \MT@permute
```


... which we can now define to be $\langle\#3\rangle$. Here, as elsewhere, we have to make the definitions global, since they will occur inside a group.

```
3445 \MT@gdef@n{MT@pr@c@MT@pr@c@name}{#3}%
3446 \MT@end@catcodes
3447 }
3448 </pdfTEX-def|XETEX-def|LUATEX-def>
```

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

```
3449 <*pdfTEX-def|LUATEX-def>
3450 \def\SetExpansion{%
3451 \MT@begin@catcodes
3452 \MT@SetExpansion
3453 }
```

`\MT@SetExpansion`

```
\MT@ex@c@name 3454 \newcommand*MT@SetExpansion[3] [] {%
\MT@extra@context 3455 \letMT@extra@context\empty
3456 \MT@set@named@keys{MT@ex@c}{#1}%
\MT@permutelist 3457 \MT@ifdefined@n@T{MT@ex@c@MT@ex@c@name @factor}{%
3458 \ifnum\c@name MT@ex@c@MT@ex@c@name @factor\endc@name > \@m
3459 \MT@warning@n1{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
3460 too large in list\MessageBreak `MT@ex@c@name'. Setting it to the
3461 maximum of 1000}%
3462 \MT@gl@et@nc{MT@ex@c@MT@ex@c@name @factor}\@m
3463 \fi
3464 }%
3465 <debug>\MT@dinfo{1}{creating expansion list `MT@ex@c@name'}%
3466 \def\MT@permutelist{ex@c}%
3467 \setkeys{MT@cfg}{#2}%
3468 \MT@permute
3469 \MT@gdef@n{MT@ex@c@MT@ex@c@name}{#3}%
3470 \MT@end@catcodes
3471 }
```

`\SetTracking`

```
3472 \def\SetTracking{%
3473 \MT@begin@catcodes
3474 \MT@SetTracking
3475 }
```

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

```
3476 \newcommand*MT@SetTracking[3] [] {%
3477 \letMT@extra@context\empty
3478 \MT@set@named@keys{MT@tr@c}{#1}%
3479 <debug>\MT@dinfo{1}{creating tracking list `MT@tr@c@name'}%
3480 \def\MT@permutelist{tr@c}%
3481 \setkeys{MT@cfg}{#2}%
3482 \MT@permute
3483 \KV@sp@def\@tempa{#3}%
3484 \MT@ifempty\@tempa\relax{%
3485 \MT@ifint\@tempa
3486 {\MT@xdef@n{MT@tr@c@MT@tr@c@name}{\@tempa}}%
3487 {\MT@warning{Value `@tempa' is not a number in\MessageBreak
3488 tracking set `MT@curr@set@name'}}}%
3489 \MT@end@catcodes
3490 }
3491 </pdfTEX-def|LUATEX-def>
```

`\SetExtraSpacing`

```
3492 <*pdfTEX-def>
3493 \def\SetExtraSpacing{%
3494 \MT@begin@catcodes
3495 \MT@SetExtraSpacing
```

```

3496 }

\MT@SetExtraSpacing
  \MT@sp@c@name 3497 \newcommand*\MT@SetExtraSpacing[3] [] {%
\MT@extra@context 3498 \let\MT@extra@context\@empty
3499 \MT@set@named@keys{MT@sp@c}{#1}%
  \MT@permutelist 3500 (debug)\MT@dinfo{1}{creating spacing list `~\MT@sp@c@name'}%
3501 \def\MT@permutelist{sp@c}%
3502 \setkeys{MT@cfg}{#2}%
3503 \MT@permute
3504 \MT@gdef@n{MT@sp@c@~\MT@sp@c@name}{#3}%
3505 \MT@end@catcodes
3506 }

\SetExtraKerning
3507 \def\SetExtraKerning{%
3508 \MT@begin@catcodes
3509 \MT@SetExtraKerning
3510 }

\MT@SetExtraKerning
  \MT@kn@c@name 3511 \newcommand*\MT@SetExtraKerning[3] [] {%
\MT@extra@context 3512 \let\MT@extra@context\@empty
3513 \MT@set@named@keys{MT@kn@c}{#1}%
  \MT@permutelist 3514 (debug)\MT@dinfo{1}{creating kerning list `~\MT@kn@c@name'}%
3515 \def\MT@permutelist{kn@c}%
3516 \setkeys{MT@cfg}{#2}%
3517 \MT@permute
3518 \MT@gdef@n{MT@kn@c@~\MT@kn@c@name}{#3}%
3519 \MT@end@catcodes
3520 }
3521 (/pdfTeX-def)

\MT@set@named@keys We first set the name (if specified), then remove it from the list, and set the
\MT@options remaining keys.
3522 (*package)
3523 \def\MT@set@named@keys#1#2{%
3524 \def\x##1name=#2,##3\@nil{%
3525 \setkeys{#1}{name=#2}%
3526 \gdef\MT@options{##1##3}%
3527 \MT@rem@from@clist{name=}\MT@options
3528 }%
3529 \x#2,name=,\@nil
3530 \@expandtwoargs\setkeys{#1}\MT@options
3531 }

\MT@define@code@key Define the keys for the configuration lists (which are setting the codes, in pdfTEX
speak).
3532 \def\MT@define@code@key#1#2{%
3533 \define@key{MT@#2}{#1} [] {%
3534 \@tempcnta=\@ne
3535 \MT@map@clist@n{##1}{%
3536 \KV@sp@def\MT@val{###1}%

Here, too, we allow for something like 'bf*'. It will be expanded immediately.
3537 \MT@get@highlevel{#1}%
3538 \MT@edef@n{MT@temp#1\the\@tempcnta}{\MT@val}%
3539 \advance\@tempcnta \@ne
3540 }%
3541 }%
3542 }

\MT@define@code@key@family Remove fontspec's internal feature counter.
3543 \def\MT@define@code@key@family#1{%

```

```

3544 \define@key{MT@#1}{family}[]{%
3545 \@tempcnta=\@ne
3546 \MT@map@clist@n{##1}{%
3547 \KV@sp@def\MT@val{###1}%
3548 \MT@get@highlevel{family}%
3549 \ifMT@fontspec
3550 \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
3551 \fi
3552 \MT@edef@n{MT@tempfamily\the\@tempcnta}{\MT@val}%
3553 \advance\@tempcnta \@ne
3554 }%
3555 }%
3556 }

```

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

```

3557 \def\MT@define@code@key@size#1{%
3558 \define@key{MT@#1}{size}[]{%
3559 \MT@map@clist@n{##1}{%
3560 \KV@sp@def\MT@val{###1}%
3561 \expandafter\MT@get@range\MT@val--\@nil
3562 \ifx\MT@val\relax \else
3563 \MT@exp@cs\MT@xadd{MT@tempsize}%
3564 {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
3565 \fi
3566 }%
3567 }%
3568 }

```

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

```

3569 \def\MT@define@code@key@font#1{%
3570 \define@key{MT@#1}{font}[]{%
3571 \MT@map@clist@n{##1}{%
3572 \KV@sp@def\MT@val{###1}%
3573 \MT@ifstreq\MT@val*\def\MT@val{*/**/*/*}\relax
3574 \expandafter\MT@get@font@and@size\MT@val////\@nil
3575 \ifMT@fontspec
3576 \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
3577 \fi
3578 \MT@xdef@n{MT@MT@permutelist @\@tempb\MT@extra@context}%
3579 {\csname MT@MT@permutelist @name\endcsname}%
3580 <debug> \MT@dinfo@n1{1}{initialising: use list for font \@tempb=\MT@val
3581 <debug> \ifx\MT@extra@context\empty\else\MessageBreak
3582 <debug> (context: \MT@extra@context)\fi}%
3583 \MT@exp@cs\MT@xaddb
3584 {MT@MT@permutelist @\@tempb\MT@extra@context @sizes}%
3585 {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
3586 }%
3587 }%
3588 }

```

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

```

3589 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
3590 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
3591 }

3592 \MT@define@code@key{encoding}{cfg}
3593 \MT@define@code@key{family}{cfg}
3594 \MT@define@code@key{series}{cfg}
3595 \MT@define@code@key{shape}{cfg}
3596 \MT@define@code@key@size{cfg}
3597 \MT@define@code@key@font{cfg}

```

`\MT@define@opt@key`

```

3598 \def\MT@define@opt@key#1#2{%
3599 \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax%

```

```

3600 \MT@xdef@n{MT@#1@c@MT@curr@set@name @#2}{##1}}%
3601 }

```

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

```

3602 \newcount\MT@listname@count
3603 \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).

```

3604 \define@key{MT@#1@c}{name} [] {%
3605 \MT@ifempty{##1}}%
3606 \MT@ifdefined@n@TF{MT@#1@c@MT@curr@file/\the\inputlineno}{%
3607 \global\advance\MT@listname@count\@ne
3608 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
3609 (\number\MT@listname@count)}%
3610 }{%
3611 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
3612 }%
3613 }{%
3614 \MT@edef@n{MT@#1@c@name}{##1}%
3615 \MT@ifdefined@n@T{MT@#1@c@\csname MT@#1@c@name\endcsname}{%
3616 \MT@warning{Redefining \@nameuse{MT@abbr@#1} list ` \@nameuse{MT@#1@c@name}' }%
3617 }%
3618 }%
3619 \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
3620 }%
3621 \MT@define@opt@key{#1}{load}%
3622 \MT@define@opt@key{#1}{factor}%
3623 \MT@define@opt@key{#1}{preset}%
3624 \MT@define@opt@key{#1}{inputenc}%

```

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

```

3625 \define@key{MT@#1@c}{context} [] {\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
3626 }
3627 /package

```

Automatically enable font copying if we find a protrusion or expansion context. After the preamble, check whether font copying is enabled. For older pdfTeX versions, disallow. It also works with LuaTeX 0.30 or newer.

```

3628 (*pdfTeX-def|luatex-def)
3629 (pdfTeX-def)\MT@requires@pdfTeX7{
3630 \define@key{MT@ex@c}{context} [] {%
3631 \MT@ifempty{##1}\relax{%
3632 \MT@gl@t\MT@copy@font\MT@copy@font@
3633 \def\MT@extra@context{##1}%
3634 }%
3635 }
3636 \MT@addto@setup{%
3637 \define@key{MT@ex@c}{context} [] {%
3638 \ifx\MT@copy@font\MT@copy@font@
3639 \MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}%
3640 \else
3641 \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
3642 Ignoring `context' key\on@line}%
3643 {Either move the settings inside the preamble,\MessageBreak
3644 or load the package with the `copyfonts' option.}%
3645 \fi
3646 }%
3647 }

```

Protrusion contexts *might* also work without copying the font, so we don't issue an error but only a warning. The problem is that pdfTeX only allows one set of protrusion factors for a given font within one paragraph (those that are in effect at

the end of the paragraph will be in effect for the whole paragraph). When different fonts are loaded – like in the example with the footnote markers – we don't need to copy the fonts.

```

3648 \define@key{MT@pr@c}{context}[]{%
3649   \MT@ifempty{#1}\relax{%
3650     \MT@gllet\MT@copy@font\MT@copy@font@
3651     \def\MT@extra@context{#1}%
3652   }%
3653 }
3654 \MT@addto@setup{%
3655   \define@key{MT@pr@c}{context}[]{%
3656     \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
3657     \ifx\MT@copy@font\MT@copy@font@ \else
3658       \MT@warning@nl{If protrusion contexts don't work as expected,
3659         \MessageBreak load the package with the `copyfonts' option}%
3660     \fi
3661   }%
3662 }
3663 </pdfTeX-def|LaTeX-def>
3664 <*pdfTeX-def>
3665 }{
3666   \define@key{MT@ex@c}{context}[]{%
3667     \MT@error{Expansion contexts only work with pdfTeX 1.40.4\MessageBreak
3668       or later. Ignoring `context' key\on@line}%
3669     {Upgrade pdfTeX.}%
3670   }
3671 </pdfTeX-def>
3672 <*pdfTeX-def|xTeX-def>
3673   \define@key{MT@pr@c}{context}[]{%
3674     \MT@error{Protrusion contexts only work with pdfTeX
3675       1.40.4\MessageBreak or later.
3676     \MessageBreak or LaTeX.
3677     Ignoring `context' key\on@line}%
3678   <pdfTeX-def> {Upgrade pdfTeX.}%
3679   <xTeX-def> {Use pdfTeX or LaTeX.}%
3680   }
3681 </pdfTeX-def|xTeX-def>
3682 <pdfTeX-def>

```

\MT@warn@nodim

```

3683 <*package>
3684 \def\MT@warn@nodim#1{%
3685   \MT@warning{`\@tempa' is not a dimension.\MessageBreak
3686     Ignoring it and setting values relative to\MessageBreak #1}%
3687 }
3688 </package>

```

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

```

3689 <*pdfTeX-def|xTeX-def|LaTeX-def>
3690 \define@key{MT@pr@c}{unit}[character]{%
3691   \MT@gllet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
3692   \def\@tempa{#1}%
3693   \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.

```

3694   \MT@ifdimen\@tempa
3695     {\MT@gllet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
3696     {\MT@warn@nodim{character widths}}%
3697   }%
3698 }
3699 </pdfTeX-def|xTeX-def|LaTeX-def>

```

Tracking may only be relative to a dimension.

```

3700 <*pdf $\textit{tex-def}$ | $\textit{luatex-def}$ >
3701 \define@key{MT@tr@ $\textit{c}$ }{unit}[1em]{%
3702   \MT@glet@nc{MT@tr@ $\textit{c}$ @\MT@curr@set@name @unit}\@empty
3703   \def\@tempa{#1}%
3704   \MT@ifdimen\@tempa
3705     {\MT@glet@nc{MT@tr@ $\textit{c}$ @\MT@curr@set@name @unit}\@tempa}%
3706     {\MT@warn@nodim{1em}%
3707       \MT@gdef@n{MT@tr@ $\textit{c}$ @\MT@curr@set@name @unit}{1em}}%
3708 }
3709 </pdf $\textit{tex-def}$ | $\textit{luatex-def}$ >

```

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

```

3710 <*pdf $\textit{tex-def}$ >
3711 \MT@map@clist@n{sp,kn}{%
3712   \define@key{MT@#1@ $\textit{c}$ }{unit}[space]{%
3713     \MT@glet@nc{MT@#1@ $\textit{c}$ @\MT@curr@set@name @unit}\@empty
3714     \def\@tempa{##1}%
3715     \MT@ifstre $\textit{q}$ \@tempa{character}\relax{%
3716       \MT@glet@nc{MT@#1@ $\textit{c}$ @\MT@curr@set@name @unit}\m@ne
3717       \MT@ifstre $\textit{q}$ \@tempa{space}\relax{%
3718         \MT@ifdimen\@tempa
3719         {\MT@glet@nc{MT@#1@ $\textit{c}$ @\MT@curr@set@name @unit}\@tempa}%
3720         {\MT@warn@nodim{width of space}}%
3721       }%
3722     }%
3723   }%
3724 }
3725 </pdf $\textit{tex-def}$ >

```

The first argument to \SetExpansion accepts some more options.

```

3726 <*pdf $\textit{tex-def}$ | $\textit{luatex-def}$ >
3727 \MT@map@clist@n{stretch,shrink,step}{%
3728   \define@key{MT@ex@ $\textit{c}$ }{#1}[]{%
3729     \MT@ifempty{##1}\relax{%
3730       \MT@ifint{##1}{%

```

A space terminates the number.

```

3731       \MT@gdef@n{MT@ex@ $\textit{c}$ @\MT@curr@set@name @#1}{##1 }%
3732     }{%
3733       \MT@warning{%
3734         Value `##1' for option `#1' is not a number.\MessageBreak
3735         Ignoring it}%
3736     }%
3737   }%
3738 }%
3739 }
3740 \define@key{MT@ex@ $\textit{c}$ }{auto}[true]{%
3741   \def\@tempa{#1}%
3742   \csname if\@tempa\endcsname

```

Don't use autoexpand for pdf $\textit{T}_{\textit{E}}\textit{X}$ version older than 1.20.

```

3743 <pdf $\textit{tex-def}$ > \MT@requires@pdf $\textit{t}e\textit{x}4$ %
3744 <luat $\textit{ex-def}$ > \MT@requires@luat $\textit{e}x3$ \relax
3745   {\MT@gdef@n{MT@ex@ $\textit{c}$ @\MT@curr@set@name @auto}{autoexpand}}%
3746 <pdf $\textit{tex-def}$ > {\MT@warning{pdf $\textit{t}e\textit{x}$  too old for automatic font expansion}}%
3747   \else
3748 <pdf $\textit{tex-def}$ > \MT@requires@pdf $\textit{t}e\textit{x}4$ %
3749 <*luat $\textit{ex-def}$ >
3750   \MT@requires@luat $\textit{e}x3$ {%
3751     \MT@warning{Non-automatic font expansion doesn't work with\MessageBreak
3752       luat $\textit{e}x$ }}%
3753 </luat $\textit{ex-def}$ >
3754   {\MT@glet@nc{MT@ex@ $\textit{c}$ @\MT@curr@set@name @auto}\@empty}%
3755 <pdf $\textit{tex-def}$ > \relax
3756 \fi

```

3757 }

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

```
3758 \MT@define@opt@key{tr}{spacing}
3759 \MT@define@opt@key{tr}{outerspacing}
3760 \MT@define@opt@key{tr}{outerkerning}
```

Which ligatures should be disabled?

```
3761 \define@key{MT@tr@c}{noligatures} []%
3762   {\MT@xdef@n{MT@tr@c@MT@curr@set@name @noligatures}{#1}}
3763 \define@key{MT@tr@c}{outer spacing} [] {\setkeys{MT@tr@c}{outerspacing={#1}}}
3764 \define@key{MT@tr@c}{outer kerning} [] {\setkeys{MT@tr@c}{outerkerning={#1}}}
3765 \define@key{MT@tr@c}{noligatures} [] {\setkeys{MT@tr@c}{noligatures={#1}}}
3766 </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, and to specify an input encoding.

`\MT@extra@inputenc`

```
3767 <*>package
3768 \renewcommand*\DeclareCharacterInheritance[1] [] [%
3769   \let\MT@extra@context\@empty
3770   \let\MT@extra@inputenc\@undefined
3771   \let\MT@inh@feat\@empty
3772   \setkeys{MT@inh@}{#1}%
3773   \MT@begin@catcodes
3774   \MT@set@inh@list
3775 }
```

`\MT@set@inh@list`

Safe category codes.

```
3776 \def\MT@set@inh@list#1#2{%
3777   \MT@ifempty\MT@inh@feat{%
3778     \MT@map@clist@c\MT@features{{\MT@declare@char@inh{##1}{#1}{#2}}}%
3779   }%
3780   \MT@map@clist@c\MT@inh@feat{{%
3781     \KV@sp@def\@tempa{##1}%
3782     \MT@ifempty\@tempa\relax{%
3783       \MT@exp@one@n\MT@declare@char@inh
3784       {\csname MT@rbba@\@tempa\endcsname}{#1}{#2}%
3785     }%
3786   }}%
3787 }%
3788 \MT@end@catcodes
3789 }
```

The keys for the optional argument.

```
3790 \MT@map@clist@c\MT@features@long{%
3791   \define@key{MT@inh@}{#1} [] {\edef\MT@inh@feat{\MT@inh@feat#1,}}
3792 \define@key{MT@inh@}{inputenc} {\def\MT@extra@inputenc{#1}}
```

`\MT@declare@char@inh`

The lists cannot be given a name by the user.

```
3793 \def\MT@declare@char@inh#1#2#3{%
3794   \MT@edef@n{MT@#1@inh@name}%
3795   {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr#1})}%
```

```

3796 \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
3797 \MT@ifdefined@c@T\MT@extra@inputenc{%
3798   \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
3799 (debug)\MT@edinfo{1}{creating inheritance list ` \@nameuse{MT@#1@inh@name}' }%
3800 \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
3801 \def\MT@permutelist{#1@inh}%
3802 \setkeys{MT@inh}{#2}%
3803 \MT@permute
3804 }

```

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

```

3805 \MT@define@code@key{encoding}{inh}
3806 \MT@define@code@key{family} {inh}
3807 \MT@define@code@key{series} {inh}
3808 \MT@define@code@key{shape} {inh}
3809 \MT@define@code@key{size} {inh}
3810 \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`).

```

3811 \def\MT@inh@do#1,{%
3812   \ifx\relax#1\empty \else
3813     \MT@inh@split #1==\relax
3814     \expandafter\MT@inh@do
3815   \fi
3816 }

```

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

```

3817 (/package)
3818 (*pdftex-def|xetex-def|luatex-def)
3819 \def\MT@inh@split#1=#2=#3\relax{%
3820   \def@tempa{#1}%
3821   \ifx@tempa\empty \else
3822     \MT@get@slot
3823 (pdftex-def|luatex-def) \ifnum\MT@char > \m@ne
3824 (xetex-def) \ifx\MT@char\empty\else
3825     \let\MT@val\MT@char
3826     \MT@map@clist@n{#2}{%
3827       \def@tempa{##1}%
3828       \ifx@tempa\empty \else
3829         \MT@get@slot
3830 (pdftex-def|luatex-def) \ifnum\MT@char > \m@ne
3831 (xetex-def) \ifx\MT@char\empty\else
3832       \MT@exp@cs\MT@xadd{MT@inh\MT@listname @\MT@val @}{\MT@char}}%
3833     \fi
3834   \fi
3835 }%
3836 (debug)\MT@edinfo{n1}{children of #1 (\MT@val):
3837 (debug) \nameuse{MT@inh\MT@listname @\MT@val @}}%
3838 \fi
3839 \fi
3840 }
3841 (/pdftex-def|xetex-def|luatex-def)

```


14.3.7 Permutation

`\MT@permute` Calling `\MT@permute` will define commands for all permutations of the specified font attributes of the form `\MT@⟨list type⟩@⟨encoding⟩/⟨family⟩/⟨series⟩/⟨shape⟩/⟨|*⟩` to be the expansion of `\MT@⟨list type⟩@name`, i.e., the name of the currently defined list. `\MT@permute@@` Size ranges are held in a separate macro called `\MT@⟨list type⟩@⟨font axes⟩@sizes`, which in turn contains the respective `⟨list name⟩s` attached to the ranges.

```

3842 *package
3843 \def\MT@permute{%
3844   \let\MT@cnt@encoding\@ne
3845   \MT@permute@

  Undefine commands for the next round.
3846   \MT@map@tlist@n{{encoding}{family}{series}{shape}}\MT@permute@reset
3847   \MT@glet\MT@tempsize\@undefined
3848 }
3849 \def\MT@permute@{%
3850   \let\MT@cnt@family\@ne
3851   \MT@permute@@
3852   \MT@increment\MT@cnt@encoding
3853   \MT@ifdefined@n@T{MT@tempencoding\MT@cnt@encoding}%
3854   \MT@permute@
3855 }
3856 \def\MT@permute@@{%
3857   \let\MT@cnt@series\@ne
3858   \MT@permute@@@
3859   \MT@increment\MT@cnt@family
3860   \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
3861   \MT@permute@@
3862 }
3863 \def\MT@permute@@@{%
3864   \let\MT@cnt@shape\@ne
3865   \MT@permute@@@@
3866   \MT@increment\MT@cnt@series
3867   \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
3868   \MT@permute@@@@
3869 }
3870 \def\MT@permute@@@@{%
3871   \MT@permute@@@@@
3872   \MT@increment\MT@cnt@shape
3873   \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
3874   \MT@permute@@@@
3875 }

```

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

```

3876 \def\MT@permute@@@@@{%
3877   \MT@permute@define(encoding)%
3878   \ifMT@document
3879     \ifx\MT@tempencoding\@empty \else
3880       \MT@ifdefined@n@TF{T@\MT@tempencoding}\relax
3881       {\expandafter\expandafter\expandafter\@gobble}%
3882     \fi
3883   \fi
3884   \MT@permute@@@@@
3885 }

```

```

\MT@permute@@@@@@
3886 \def\MT@permute@@@@@@{%
3887   \MT@permute@define{family}%
3888   \MT@permute@define{series}%
3889   \MT@permute@define{shape}%
3890   \edef\@tempa{\MT@tempencoding

```

```

3891         /\MT@tempfamily
3892         /\MT@tempseries
3893         /\MT@tempshape
3894         /\MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

3895 \MT@iffstreq\@tempa{////}\relax{%
3896   \ifx\MT@tempencoding\empty
3897     \MT@warning{%
3898       You have to specify an encoding for\MessageBreak
3899       \@nameuse{MT@abbr@MT@permutelist} list
3900       ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
3901       Ignoring it}%
3902   \else
3903     \MT@ifdefined@c@TF\MT@tempsize{%

```

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

```

3904     \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%
3905       \MT@map@tlist@c\MT@tempsize\MT@check@rlist
3906     }%
3907     \MT@exp@cs\MT@xaddb
3908     {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
3909     \MT@tempsize
3910 <debug> \MT@dinfo@n{1}{initialising: use list for font \@tempa,\MessageBreak
3911 <debug>       sizes: \csname MT@MT@permutelist @\@tempa\MT@extra@context
3912 <debug>       @sizes\endcsname}%
3913     }{%

```

Only one list can apply to a given combination. But we don't warn if the overridden list is to be loaded by the current one.

```

3914     \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context}{%
3915       \MT@ifstreq{\csname MT@MT@permutelist @\@tempa\MT@extra@context\endcsname}%
3916       {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
3917       \relax{%
3918         \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
3919           ~\@nameuse{MT@MT@permutelist @name}' will\MessageBreak override
3920           list ~\@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
3921           for \MessageBreak font ~\@tempa'}%
3922       }%
3923     }%
3924 <debug> \MT@dinfo@n{1}{initialising: use list for font \@tempa
3925 <debug>       \ifx\MT@extra@context\empty\else\MessageBreak
3926 <debug>       (context: \MT@extra@context)\fi}%
3927     }%
3928     \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}%
3929     {\csname MT@MT@permutelist @name\endcsname}%
3930     \fi
3931   }%
3932 }

```

\MT@permute@define Define the commands.

```

3933 \def\MT@permute@define#1{%
3934   \@tempcnta=\csname MT@cnt@#1\endcsname\relax
3935   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3936   {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
3937   {\MT@let@nc{MT@temp#1}\@empty}%
3938 }

```

\MT@permute@reset Reset the commands.

```

3939 \def\MT@permute@reset#1{%
3940   \@tempcnta=\@ne
3941   \MT@loop
3942   \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined

```

```

3943     \advance\@tempcnta\@ne
3944     \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3945         \iftrue
3946         \iffalse
3947     \MT@repeat
3948 }

\MT@check@rlist    For every new range item in \MT@tempsize, check whether it overlaps with ranges
                   in the existing list.
3949 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

\MT@check@rlist@   Define the current new range and ...
3950 \def\MT@check@rlist@#1#2#3{%
3951     \def\@tempb{#1}%
3952     \def\@tempc{#2}%
3953     \MT@if@false
3954     \MT@exp@cs\MT@map@tlist@
3955     {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
3956     \MT@check@range
3957 }

\MT@check@range    ... recurse through the list of existing ranges.
3958 \def\MT@check@range#1{\expandafter\MT@check@range@ #1}

\MT@check@range@   \@tempb and \@tempc are lower resp. upper bound of the new range, <#1> and <#2>
                   those of the existing range. <#3> is the list name.
3959 \def\MT@check@range@#1#2#3{%
3960     \MT@ifdim{#2}=\m@ne{%
3961         \MT@ifdim\@tempc=\m@ne{%



- Both items are simple sizes.


3962         \MT@ifdim\@tempb={#1}\MT@iftrue\relax
3963     }{%



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


3964         \MT@ifdim\@tempb>{#1}\relax{%
3965             \MT@ifdim\@tempc>{#1}{%
3966                 \MT@iftrue
3967                 \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
3968             }\relax
3969         }%
3970     }%
3971 }{%



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


3973     \MT@ifdim\@tempb<{#2}{%
3974         \MT@ifdim\@tempb<{#1}\relax\MT@iftrue
3975     }\relax
3976 }{%



- Both items are ranges.


3977     \MT@ifdim\@tempb<{#2}{%
3978         \MT@ifdim\@tempc>{#1}{%
3979             \MT@iftrue
3980             \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
3981         }\relax
3982     }\relax
3983 }%
3984 }%
3985 \ifMT@if@

```

```

3986 \MT@ifstreq{#3}%
3987   {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
3988   \relax}%
3989   \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
3990   ~\@nameuse{MT@MT@permutelist @name}' will override\MessageBreak
3991   list ~#3' for font \@tempa,\MessageBreak size \@tempb}%
3992 }%

```

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

```

3993   \expandafter\MT@tlist@break
3994 \fi
3995 }

```

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 3996 \newif\ifMT@opt@expansion
\ifMT@opt@DVI 3997 \newif\ifMT@opt@auto
3998 \newif\ifMT@opt@DVI

\MT@optwarn@admissible   Some warnings.
3999 \def\MT@optwarn@admissible#1#2{%
4000   \MT@warning@n1{~#1' is not an admissible value for option\MessageBreak
4001   ~#2'. Assuming ~false'}%
4002 }

\MT@optwarn@nan
4003 /package
4004 <package|letterspace
4005 plain\MT@requires@latex1{
4006 \def\MT@optwarn@nan#1#2{%
4007   \MT@warning@n1{Value ~#1' for option ~#2' is not a\MessageBreak number.
4008   Using default value of \number\@nameuse{MT@#2@default}}%
4009 }
4010 plain}\relax
4011 /package|letterspace
4012 *package

\MT@opt@def@set
4013 \def\MT@opt@def@set#1{%
4014   \MT@ifdefined@n@TF{MT@\@tempb @set@MT@val}{%
4015     \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
4016   }{%
4017     \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
4018     \MT@warning@n1{The #1 set ~\MT@val' is undeclared.\MessageBreak
4019     Using set ~\@nameuse{MT@\@tempb @setname}' instead}%
4020   }%
4021 }

expansion and protrusion may be true, false, compatibility, nocompatibility
and/or a set name.
4022 \MT@map@clist@n{protrusion,expansion}{%
4023   \define@key{MT}{#1}[true]{%
4024     \csname MT@opt@#1true\endcsname
4025     \MT@map@clist@n{##1}{%
4026       \KV@sp@def\MT@val{###1}%
4027       \MT@ifempty\MT@val\relax%
4028       \csname MT@#1true\endcsname
4029       \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4030       \MT@ifstreq\MT@val{true}\relax

```

```

4031     {%
4032     \MT@ifstreq\MT@val{false}{%
4033     \csname MT@#1false\endcsname
4034     }{%
4035     \MT@ifstreq\MT@val{compatibility}{%
4036     \MT@let@nc{MT@\@tempb @level}\@ne
4037     }{%
4038     \MT@ifstreq\MT@val{nocompatibility}{%
4039     \MT@let@nc{MT@\@tempb @level}\tw@
4040     }{%

```

If everything failed, it should be a set name.

```

4041     \MT@opt@def@set{#1}%
4042     }%
4043     }%
4044     }%
4045     }%
4046     }%
4047     }%
4048     }%
4049 }

```

activate is a shortcut for protrusion and expansion.

```

4050 \define@key{MT}{activate}[true]{%
4051 \setkeys{MT}{protrusion={#1}}%
4052 \setkeys{MT}{expansion={#1}}%
4053 }

```

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

```

4054 \MT@map@clist@n{spacing,kerning,tracking}{%
4055 \define@key{MT}{#1}[true]{%
4056 \MT@map@clist@n{##1}{%
4057 \KV@sp@def\MT@val{###1}%
4058 \MT@ifempty\MT@val\relax{%
4059 \csname MT@#1true\endcsname
4060 \MT@ifstreq\MT@val{true}\relax
4061 }%
4062 \MT@ifstreq\MT@val{false}{%
4063 \csname MT@#1false\endcsname
4064 }{%
4065 \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4066 \MT@opt@def@set{#1}%
4067 }%
4068 }%
4069 }%
4070 }%
4071 }%
4072 }

```

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

```

4073 \def\MT@def@bool@opt#1#2{%
4074 \define@key{MT}{#1}[true]{%
4075 \def\@tempa{##1}%
4076 \MT@ifstreq\@tempa{true}\relax{%
4077 \MT@ifstreq\@tempa{false}\relax{%
4078 \MT@optwarn@admissible{##1}{#1}%
4079 \def\@tempa{false}%
4080 }%
4081 }%
4082 #2%
4083 }%
4084 }

```

Boolean options that only set the switch.

```

4085 \MT@map@clist@n{draft,selected,babel}{%
4086   \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
4087 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotru}

```

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

```

4088 </package>
4089 <*pdftex-def|luatex-def|xetex-def>
4090 <luatex-def>\MT@requires@luatex4{\let\pdfoutput\outputmode}\relax
4091 \MT@def@bool@opt{DVIPDF}{%
4092   \csname if\@tempa\endcsname
4093 <*pdftex-def|luatex-def>
4094   \ifnum\pdfoutput>\z@ \MT@opt@DVIPDF \fi
4095   \pdfoutput\z@
4096   \else
4097   \ifnum\pdfoutput<\@ne \MT@opt@DVIPDF \fi
4098   \pdfoutput\@ne
4099 </pdftex-def|luatex-def>
4100 <xetex-def> \MT@warning@n1{Ignoring `DVIPDF' option}%
4101   \fi
4102 }
4103 </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.

```

4104 <*package>
4105 \MT@def@bool@opt{defersetup}{%
4106   \csname if\@tempa\endcsname \else
4107     \AtEndOfPackage{%
4108       \MT@setup@
4109       \let\MT@setup@\@empty
4110       \let\MT@addto@setup\@firstofone
4111     }%
4112   \fi
4113 }
4114 </package>

```

`copyfonts` will copy all fonts before setting them up. This allows protrusion and expansion with different parameters. This option is also *undocumented* in the hope that we can always find out automatically whether it's required. It also works with Lua_T_EX 0.30 or newer.

```

4115 <*pdftex-def|luatex-def>
4116 <pdftex-def>\MT@requires@pdftex7{
4117   \MT@def@bool@opt{copyfonts}{%
4118     \csname if\@tempa\endcsname
4119     \MT@gl@et\MT@copy@font\MT@copy@font@
4120     \else
4121     \MT@gl@et\MT@copy@font\relax
4122     \fi
4123   }
4124 <pdftex-def>}{
4125 </pdftex-def|luatex-def>
4126 <*pdftex-def|xetex-def>
4127   \MT@def@bool@opt{copyfonts}{%
4128     \csname if\@tempa\endcsname
4129     \MT@error
4130 <pdftex-def> {The pdftex version you are using is too old\MessageBreak
4131 <pdftex-def> to use the `copyfonts' option}{Upgrade pdftex.}%

```

```

4132 <xetex-def>          {The `copyfonts' option does not work with xetex}
4133 <xetex-def>          {Use pdftex or luatex instead.}%
4134   \fi
4135 }
4136 <pdftex-def>
4137 </pdftex-def|xetex-def>

```

`final` is the opposite to `draft`.

```

4138 <*package>
4139 \MT@def@bool@opt{final}{%
4140   \csname if@tempa\endcsname
4141   \MT@draftfalse
4142   \else
4143   \MT@drafttrue
4144   \fi
4145 }

```

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

```

4146 \define@key{MT}{verbose}[true]{%
4147   \let\MT@vinfo\MT@info@n1
4148   \def\@tempa{#1}%
4149   \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

4150   \MT@ifstreq\@tempa{errors}{%
4151     \let\MT@warning \MT@warn@err
4152     \let\MT@warning@n1\MT@warn@err
4153   }{%
4154     \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

4155     \MT@ifstreq\@tempa{silent}{%
4156       \let\MT@warning \MT@info
4157       \let\MT@warning@n1\MT@info@n1
4158     }{%
4159       \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
4160     }%
4161   }%
4162 }%
4163 }
4164 </package>

```

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

```

4165 <*package|letterspace>
4166 <plain>\MT@requires@latex1{
4167   \MT@map@clist@n{%
4168     <package> stretch,shrink,step,%
4169     letterspace}{%
4170     \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
4171     \def\@tempa{##1 }%

```

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

```

4172     \MT@ifint\@tempa
4173     {\MT@edef@n{MT@#1}{\@tempa}}%
4174     {\MT@optwarn@nan{##1}{#1}}%
4175   }%
4176 }
4177 <plain>}\relax
4178 </package|letterspace>

```

`factor` will define the protrusion factor only.

```

4179 <*package>
4180 \define@key{MT}{factor}[\MT@factor@default]{%
4181   \def\@tempa{#1 }%
4182   \MT@ifint\@tempa

```

```

4183   {\edef\MT@pr@factor{\@tempa}}
4184   {\MT@optwarn@nan{#1}{factor}}%
4185 }

```

Unit for protrusion codes.

```

4186 \define@key{MT}{unit}[character]{%
4187   \def\@tempa{#1}%
4188   \MT@ifstreq\@tempa{character}\relax{%
4189     \MT@ifdimen\@tempa
4190     {\let\MT@pr@unit\@tempa}%
4191     {\MT@warning@n1{\@tempa' is not a dimension.\MessageBreak
4192       Ignoring it and setting values relative to\MessageBreak
4193       character widths}}}%
4194 }%
4195 }

```

14.4.2 Loading the definition file

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

```

4196 \let\MT@endinput\relax
4197 \ifx\MT@engine\relax
4198   \MT@warning@n1{You don't seem to be using pdftex, luatex or xetex.\MessageBreak
4199     ~\MT@MT' only works with these engines.\MessageBreak
4200     I will quit now}
4201   \MT@clear@options
4202 \else

```

Otherwise load the engine-specific code (as strewn across this file).

```

4203   \input{microtype-\MT@engine tex.def}
4204 \fi
4205 \MT@endinput

```

14.4.3 Reading the configuration file

The package should just work if called without any options. Therefore, expansion will be switched off by default if output is DVI, since it isn't likely that expanded fonts are available. (This grows more important as modern T_EX systems have switched to the pdfT_EX engine even for DVI output, so that the user might not even be aware of the fact that she's running pdfT_EX.)

```

4206 \MT@protrusiontrue
4207 </package>
4208 <{*pdftex-def|luatex-def}
4209 \ifnum\pdfoutput<\@ne \else

```

Also, we only enable expansion by default if pdfT_EX can expand the fonts automatically.

```

4210 <pdftex-def> \MT@requires@pdftex4{
4211   \MT@expansiontrue
4212 <pdftex-def> \MT@autottrue
4213 <pdftex-def> }\relax
4214 \fi
4215 <luatex-def>\MT@autottrue
4216 </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.

```

4217 <package>
4218 \define@key{MT}{config}[]{\relax}
4219 \def\MT@get@config#1config=#2,#3\@nil{%

```



```

4220 \MT@ifempty{#2}%
4221   {\def\MT@config@file{\MT@MT.cfg}}%
4222   {\def\MT@config@file{#2.cfg}}%
4223 }
4224 \expandafter\expandafter\expandafter\MT@get@config
4225 \csname opt@\@currname.\@currxt\endcsname,config=,\@nil

```

Load the file.

```

4226 \IfFileExists{\MT@config@file}{%
4227   \MT@info@nl{Loading configuration file \MT@config@file}%
4228   \MT@begin@catcodes
4229   \let\MT@begin@catcodes\relax
4230   \let\MT@end@catcodes\relax
4231   \let\MT@curr@file\MT@config@file
4232   \input{\MT@config@file}%
4233   \endgroup
4234 }{\MT@warning@nl{%
4235   Could not find configuration file ` \MT@config@file!'\MessageBreak
4236   This will almost certainly cause undesired results.\MessageBreak
4237   Please fix your installation}%
4238 }

```

`\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).

```

4239 \def\MT@check@active@set#1{%
4240   \MT@ifdefined@n@TF{MT@#1@setname}{%
4241     \MT@info@nl{Using \nameuse{MT@abbr@#1} set ` \nameuse{MT@#1@setname}'}%
4242   }{%
4243     \MT@ifdefined@n@TF{MT@default@#1@set}{%
4244       \MT@glet@nn{MT@#1@setname}{MT@default@#1@set}%
4245       \MT@info@nl{Using default \nameuse{MT@abbr@#1} set ` \nameuse{MT@#1@setname}'}%
4246     }{%

```

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

```

4247     \MT@gdef@n{MT@#1@setname}{@}%
4248     \MT@warning@nl{No \nameuse{MT@abbr@#1} set chosen, no default set declared.
4249       \MessageBreak Using empty set}%
4250   }%
4251 }%
4252 }

```

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}
   {\gaddto@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.

```

4253 \MT@ifdefined@cc@T\MicroType@Hook{\MT@warning{%
4254   Command \string\MicroType@Hook\space is deprecated.\MessageBreak
4255   Use \string\Microtype@Hook\space instead}\MicroType@Hook}
4256 \MT@ifdefined@cc@T\Microtype@Hook\Microtype@Hook

```

14.4.5 Changing options later

`\microtypesetup`
`\MT@define@optionX`

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

```

4257 \def\microtypesetup{\setkeys{MT}}
4258 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
4259 \end{package}
4260 \ifpdf\def\microtypesetup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
4261 \def\MT@define@optionX#1#2{%
4262   \define@key{MTX}{#1}[true]{%
4263     \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4264     \MT@map@cliston{##1}{%
4265       \KV@sp@def\MT@val{###1}%
4266       \MT@ifempty\MT@val\relax{%
4267         \@tempcnta=\m@ne
4268         \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.

```

4269   \MT@checksetup{#1}{%
4270     \@tempcnta=\csname MT@\@tempb @level\endcsname
4271     \MT@vinfo{Enabling #1
4272               (level \number\csname MT@\@tempb @level\endcsname)\on@line}%
4273     }%
4274   }{%
4275     \MT@ifstreq\MT@val{false}{%
4276       \@tempcnta=\z@
4277       \MT@vinfo{Disabling #1\on@line}%
4278     }{%
4279       \MT@ifstreq\MT@val{compatibility}{%
4280         \MT@checksetup{#1}{%
4281           \@tempcnta=\@ne
4282           \MT@let@nc{MT@\@tempb @level}\@ne
4283           \MT@vinfo{Setting #1 to level 1\on@line}%
4284         }%
4285       }{%
4286         \MT@ifstreq\MT@val{nocompatibility}{%
4287           \MT@checksetup{#1}{%
4288             \@tempcnta=\tw@
4289             \MT@let@nc{MT@\@tempb @level}\tw@
4290             \MT@vinfo{Setting #1 to level 2\on@line}%
4291           }%
4292         }{\MT@error{Value '\MT@val' for key `#1' not recognised}

```

```

4293             (Use any of `true', `false', `compatibility' or
4294             `nocompatibility'.)%
4295         }%
4296     }%
4297 }%
4298 }%
4299 \ifnum\@tempcnta>\m@ne
4300 #2\@tempcnta\relax
4301 \fi
4302 }%
4303 }%
4304 }%
4305 }

```

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

```

4306 \def\MT@checksetup#1{%
4307   \csname ifMT@#1\endcsname
4308   \expandafter\@firstofone
4309   \else
4310     \MT@error{You cannot enable #1 if it was disabled\MessageBreak
4311             in the package options}{Load microtype with #1 enabled.}%
4312     \expandafter\@gobble
4313   \fi
4314 }

4315 \MT@define@optionX{protrusion}\MT@protrudechars
4316 (/pdftex-def|luatex-def|xettex-def)
4317 (*pdftex-def|luatex-def)
4318 \MT@define@optionX{expansion}\MT@adjustspacing

```

`\MT@protrudechars`

```

\MT@adjustspacing 4319 (*luatex-def)
4320 \MT@requires@luatex4{
4321   \let\pdfprotrudechars\protrudechars
4322   \let\pdfadjustspacing\adjustspacing
4323 } \relax
4324 (/luatex-def)
4325 \let\MT@protrudechars\pdfprotrudechars
4326 \let\MT@adjustspacing\pdfadjustspacing
4327 (pdftex-def|luatex-def)
4328 (*xettex-def)
4329 \let\MT@protrudechars\XeTeXprotrudechars
4330 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
4331 (xettex-def)

```

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

```

4332 (*pdftex-def|luatex-def)
4333 (pdftex-def)\MT@requires@pdftex6{
4334 (luatex-def)\MT@requires@luatex3{
4335   \def\MT@define@optionX@#1#2{%
4336     \define@key{MTX}{#1}[true]{%
4337       \MT@map@clist@n{##1}{%
4338         \KV@sp@def\MT@val{###1}%
4339         \MT@ifempty\MT@val\relax{%
4340           \@tempcnta=\m@ne
4341           \MT@ifstreq\MT@val{true}{%
4342             \MT@checksetup{#1}{%
4343               \@tempcnta=\@ne
4344               \MT@vinfo{Enabling #1\on@line}%
4345             }%
4346           }%
4347           \MT@ifstreq\MT@val{false}{%
4348             \@tempcnta=\z@
4349             \MT@vinfo{Disabling #1\on@line}%

```

```

4350         }\MT@error{Value `~\MT@val' for key `~#1' not recognised}
4351         {Use either `true' or `false'}%
4352     }%
4353 }%
4354 \ifnum\@tempcnta>\m@ne
4355     #2\relax
4356 \fi
4357 }%
4358 }%
4359 }%
4360 }

```

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

```

4361 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
4362     \else \let\MT@tracking\MT@tracking@ \fi}
4363 (pdfTeX-def) \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
4364 (pdfTeX-def) \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
4365 (pdfTeX-def) \pdfappendkern\@tempcnta}
4366 }{
4367 (pdfTeX-def|LaTeX-def)
4368 (pdfTeX-def|LaTeX-def|xTeX-def)

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

4369 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
4370 (LaTeX-def)
4371 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
4372 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
4373 (pdfTeX-def)
4374 \define@key{MTX}{activate}[true]{%
4375     \setkeys{MTX}{protrusion=#1}}%
4376 (pdfTeX-def|LaTeX-def) \setkeys{MTX}{expansion=#1}}%
4377 }
4378 (pdfTeX-def|LaTeX-def|xTeX-def)

```

`\MT@saved@setupfont` Disable everything – may be used as a temporary work-around in case setting up fonts doesn't work under certain circumstances, but only until that specific problem is fixed. This is *undocumented*, as it completely deprives us of the possibility to act – we're blind and paralysed.

```

4379 (package)
4380 \let\MT@saved@setupfont\MT@setupfont
4381 \define@key{MTX}{disable}[]{}%
4382 \MT@info{Inactivate `~\MT@MT' package}%
4383 \let\MT@setupfont\relax
4384 }
4385 \define@key{MTX}{enable}[]{}%
4386 \MT@info{Reactivate `~\MT@MT' package}%
4387 \let\MT@setupfont\MT@saved@setupfont
4388 }
4389 (package)

```

14.4.6 Processing the options

`\MT@ProcessOptionsWithKV` Parse options.

```

4390 (package|letterspace)
4391 (plain)\MT@requires@latex1{
4392 \def\MT@ProcessOptionsWithKV#1{%
4393     \let\@tempc\relax
4394     \let\MT@temp\@empty
4395 (plain) \MT@requires@latex2{
4396     \MT@map@clist@c@classoptionslist{%
4397     \def\CurrentOption{##1}%

```

```

4398     \MT@ifdefined@n@T{KV@#1@}\expandafter\MT@getkey\CurrentOption=\@nil}{%
4399         \edef\MT@temp{\MT@temp,\CurrentOption,}%
4400         \@expandtwoargs\@removeelement\CurrentOption
4401         \@unusadoptionlist\@unusadoptionlist
4402     }%
4403 }%
4404 \edef\MT@temp{\noexpand\setkeys{#1}%
4405             {\MT@temp\optionlist{\@currname.\@current}}}%

```

plain can handle package options.

```

4406 <plain>
4407 }{\edef\MT@temp{\noexpand\setkeys{#1}%
4408             {\csname usepkg@options@usepkg@pkg\endcsname}}}
4409 </plain>
4410 \MT@temp
4411 \MT@clear@options
4412 }

```

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

```

4413 \def\MT@getkey#1=#2\@nil{#1}
4414 \MT@ProcessOptionsWithKV{MT}
4415 <plain>}\relax
4416 </package|letterspace>
4417 <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).

```

4418 \MT@addto@setup{%
4419 \ifMT@draft

```

We disable most of what we've just defined in the 4419 lines above if we are running in draft mode.

```

4420 \MT@warning@n1{\`draft' option active.\MessageBreak
4421             Disabling all micro-typographic extensions.\MessageBreak
4422             This might lead to different line and page breaks}%
4423 \let\MT@setupfont\relax
4424 \renewcommand*\LoadMicrotypeFile[1]{}%
4425 \renewcommand*\microtypesetup[1]{}%
4426 \renewcommand*\microtypecontext[1]{}%
4427 \renewcommand*\lsstyle{}%
4428 \else
4429 \MT@setup@PDF
4430 \MT@setup@copies

```

Fix the font sets.

```

4431 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
4432 \MT@setup@protrusion
4433 \MT@setup@expansion
4434 \MT@setup@tracking
4435 \MT@setup@warntracking
4436 \MT@setup@spacing
4437 \MT@setup@kerning
4438 \MT@setup@noligatures
4439 }
4440 </package>

```

`\MT@setup@PDF` pdf \TeX 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!

```
4441 (*pdf $\text{tex}$ -def| $\text{luatex}$ -def)
4442 \def\MT@setup@PDF{%
4443   \MT@info@n1{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
4444     \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
4445 }
```

\backslash MT@setup@copies Working on font copies?

```
4446 \def\MT@setup@copies{%
4447   \ifx\MT@copy@font\relax\else \MT@info@n1{Using font copies for contexts}\fi
4448 }
4449 (/pdf $\text{tex}$ -def| $\text{luatex}$ -def)
4450 (* $\text{xetex}$ -def)
4451 \let\MT@setup@PDF\relax
4452 \let\MT@setup@copies\relax
4453 (/ $\text{xetex}$ -def)
```

\backslash MT@setup@protrusion Protrusion.

```
4454 (*pdf $\text{tex}$ -def| $\text{xetex}$ -def| $\text{luatex}$ -def)
4455 \def\MT@setup@protrusion{%
4456   \ifMT@protrusion
4457     \edef\MT@active@features{\MT@active@features,pr}%
4458     \MT@protrudechars\MT@pr@level
4459     \MT@info@n1{Character protrusion enabled (level \number\MT@pr@level)%
4460       \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
4461         factor: \number\MT@pr@factor\fi
4462       \ifx\MT@pr@unit@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
4463     \MT@check@active@set{pr}%
4464   \else
4465     \let\MT@protrusion\relax
4466     \MT@info@n1{No character protrusion}%
4467   \fi
4468 }
4469 (/pdf $\text{tex}$ -def| $\text{xetex}$ -def| $\text{luatex}$ -def)
```

\backslash MT@setup@expansion For DVI output, the user must have explicitly passed the expansion option to the package. Under Lua $\text{T}_{\text{E}}\text{X}$, expansion works quite differently: the glyphs will be positioned as if they were transformed, without actually being transformed. Since this could still be considered a viable option, we don't disable the feature completely, but issue a warning.

```
4470 (*pdf $\text{tex}$ -def| $\text{luatex}$ -def)
4471 \def\MT@setup@expansion{%
4472   \ifnum\pdfoutput<\@ne
4473     \ifMT@opt@expansion
4474     (* $\text{luatex}$ -def)
4475       \ifMT@expansion
4476         \MT@requires@ $\text{luatex}$ 3{%
4477           \MT@warning@n1{Font expansion doesn't work properly with  $\text{luatex}$  in\MessageBreak
4478             DVI mode: the glyphs won't be actually transformed,\MessageBreak
4479             but will only be shifted. You might want to use\MessageBreak
4480             pdf $\text{latex}$  instead. I'll continue anyway ..}%
4481           %\MT@expansionfalse
4482         }\relax
4483       \fi
4484     (/ $\text{luatex}$ -def)
4485     \else
4486       \MT@expansionfalse
4487     \fi
4488   \fi
4489   \ifMT@expansion
```

Set up the values for font expansion: if stretch has not been specified, we take the

default value of 20.

```
4490 \ifnum\MT@stretch=\m@ne
4491 \let\MT@stretch\MT@stretch@default
4492 \fi
```

If `shrink` has not been specified, it will inherit the value from `stretch`.

```
4493 \ifnum\MT@shrink=\m@ne
4494 \let\MT@shrink\MT@stretch
4495 \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 $\min(\text{stretch}, \text{shrink})/5$, rounded off, minimum value 1.

```
4496 \ifnum\MT@step=\m@ne
4497 <pdfTeX-def> \MT@requires@pdftex6{%
4498 \def\MT@step{1 }%
4499 <pdfTeX-def>
4500 }{%
4501 \ifnum\MT@stretch>\MT@shrink
4502 \ifnum\MT@shrink=\z@
4503 \@tempcnta=\MT@stretch
4504 \else
4505 \@tempcnta=\MT@shrink
4506 \fi
4507 \else
4508 \ifnum\MT@stretch=\z@
4509 \@tempcnta=\MT@shrink
4510 \else
4511 \@tempcnta=\MT@stretch
4512 \fi
4513 \fi
4514 \divide\@tempcnta 5\relax
4515 \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
4516 \edef\MT@step{\number\@tempcnta\space}%
4517 }%
4518 </pdfTeX-def>
4519 \fi
4520 \ifnum\MT@step=\z@
4521 \MT@warning@n1{The expansion step cannot be set to zero.\MessageBreak
4522 Setting it to one}%
4523 \def\MT@step{1 }%
4524 \fi
```

`\MT@auto` Automatic expansion of the font? This new feature of pdfTeX 1.20 makes the `fix` programme really usable. It must be either ‘autoexpand’ or empty (or ‘1000’ for older versions of pdfTeX). With LuaTeX, we just leave it empty, as there’s actually no difference – non-automatic font expansion doesn’t work anymore. In LuaTeX 1.0.6, the ‘autoexpand’ option seems to have been removed altogether and would trigger a warning.

```
4525 \let\MT@auto\empty
4526 \ifMT@auto
```

We turn off automatic expansion if output mode is DVI.

```
4527 <pdfTeX-def>
4528 \MT@requires@pdftex4{%
4529 \ifnum\pdfoutput<\@ne
4530 \ifMT@opt@auto
4531 \MT@error{%
4532 Automatic font expansion only works for PDF output.\MessageBreak
4533 However, you are creating a DVI file}
```

```

4534         {If you have created expanded fonts instances, remove `auto' from%
4535         \MessageBreak the package options. Otherwise, you have to switch
4536         off expansion\MessageBreak completely.}%
4537     \fi
4538     \MT@autofalse
4539 \else
4540     \def\MT@auto{autoexpand}%
4541 \fi

```

Also, if pdf_TE_X is too old.

```

4542     }{%
4543     \MT@error{%
4544     The pdftex version you are using is too old for\MessageBreak
4545     automatic font expansion}%
4546     {If you have created expanded fonts instances, remove `auto' from\MessageBreak
4547     the package options. Otherwise, you have to switch off expansion\MessageBreak
4548     completely, or upgrade pdftex to version 1.20 or newer.}%
4549     \MT@autofalse
4550     \def\MT@auto{1000 }%
4551     }%
4552 /pdfTEX-def
4553 luatex-def \MT@requires@luatex3\relax{\def\MT@auto{autoexpand}}%
4554 \else
4555 *pdfTEX-def

```

No automatic expansion.

```

4556     \MT@requires@pdfTEX4\relax{%
4557     \def\MT@auto{1000 }%
4558     }%
4559 /pdfTEX-def
4560 *luatex-def
4561     \MT@requires@luatex3{%
4562     \ifMT@opt@auto
4563     \MT@error{Non-automatic font expansion does not work with\MessageBreak
4564     luatex}{Remove `auto=false' from the package options, or use pdfTEX.}%
4565     \MT@autotrue
4566     \fi
4567     }\relax
4568 /luatex-def
4569     \fi

```

Choose the appropriate macro for selected expansion.

```

4570     \ifMT@selected
4571     \let\MT@set@ex@codes\MT@set@ex@codes@
4572     \else
4573     \let\MT@set@ex@codes\MT@set@ex@codes@n
4574     \fi

```

Filter out stretch=0, shrink=0, since it would result in a pdf_TE_X error.

```

4575     \ifnum\MT@stretch=\z@
4576     \ifnum\MT@shrink=\z@
4577     \MT@warning@nl{%
4578     Both the stretch and shrink limit are set to zero.\MessageBreak
4579     Disabling font expansion}%
4580     \MT@expansionfalse
4581     \fi
4582     \fi
4583 \fi
4584 \ifMT@expansion
4585 \edef\MT@active@features{\MT@active@features,ex}%
4586 \MT@adjustspacing\MT@ex@level
4587 \MT@info@nl{\ifMT@auto A\else Non-a\fi utomatic font expansion enabled
4588 (level \number\MT@ex@level),\MessageBreak
4589 stretch: \number\MT@stretch, shrink: \number\MT@shrink,
4590 step: \number\MT@step, \ifMT@selected\else non-\fi selected}%

```


`\MT@check@step` Check whether stretch and shrink are multiples of step.

```

4591 \def\MT@check@step##1{%
4592   \@tempcnta=\csname MT@##1\endcsname
4593   \divide\@tempcnta \MT@step
4594   \multiply\@tempcnta \MT@step
4595   \ifnum\@tempcnta=\csname MT@##1\endcsname\else
4596     \MT@warning@nl{The ##1 amount is not a multiple of step.\MessageBreak
4597       The effective maximum ##1 is \the\@tempcnta\space
4598       (step \number\MT@step)}%
4599   \fi
4600 }%
4601 \MT@check@step{stretch}%
4602 \MT@check@step{shrink}%
4603 \MT@check@active@set{ex}%

```

`\showhyphens` Inside `\showhyphens`, font expansion should be disabled. (Since 2017/01/10, the \LaTeX format contains a different version for $X_{\text{Y}}\TeX$, but since expansion doesn't work with $X_{\text{Y}}\TeX$, we don't have to bother.) Since 2019/10/01, the command is robust.

```

4604 \MT@ifdefined@n@TF{showhyphens }{%
4605   \def\MT@temp##1##2{%
4606     \expandafter\CheckCommand\csname showhyphens \endcsname[1]{##1}%
4607     \DeclareRobustCommand\showhyphens[1]{##2}%
4608   }%
4609   \def\MT@temp##1##2{%
4610     \CheckCommand*\showhyphens[1]{##1}%
4611     \gdef\showhyphens###1{##2}%
4612   }%
4613   \MT@temp
4614   {\setbox0\vbox{\color@begingroup
4615     \everypar{\parfillskip\z@skip
4616       \hsizemaxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
4617       \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}
4618   {\setbox0\vbox{\color@begingroup\pdfadjustspacing\z@
4619     \everypar{\parfillskip\z@skip
4620       \hsizemaxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
4621       \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
4622   \else
4623     \let\MT@expansion\relax
4624     \MT@info@nl{No font expansion}%
4625   \fi
4626 }
4627 </pdfTeX-def|luatex-def>
4628 <*/xetex-def>
4629 \def\MT@setup@expansion{%
4630   \ifMT@expansion
4631     \ifMT@opt@expansion
4632       \MT@error{Font expansion does not work with xetex}
4633       {Use pdfTeX or luatex instead.}%
4634   \fi
4635   \fi
4636 }
4637 </xetex-def>

```

`\MT@setup@tracking` Tracking, spacing and kerning.

```

4638 <*/pdfTeX-def|luatex-def>
4639 <pdfTeX-def>\MT@requires@pdfTeX6{%
4640 <luatex-def>\MT@requires@luatex3{%
4641   \def\MT@setup@tracking{%
4642     \ifMT@tracking
4643       \edef\MT@active@features{\MT@active@features,tr}%
4644       \MT@info@nl{Tracking enabled}%
4645       \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

4646   \ifMT@protrusion\else\MT@protrudechars\@ne\fi
4647   \else
4648     \let\MT@tracking\relax
4649     \MT@info@nl{No adjustment of tracking}%
4650   \fi
4651 }
4652 /pdftex-def|luatex-def

```

`\MT@setup@spacing`

```

4653 (*pdftex-def
4654   \def\MT@setup@spacing{%
4655     \ifMT@spacing
4656       \edef\MT@active@features{\MT@active@features,sp}%
4657       \pdfadjustinterwordglue\@ne
4658       \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.

```

4659     \MT@with@package@T{ragged2e}{%
4660       \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
4661         Adjustment of interword spacing may lead to\MessageBreak
4662         undesired results when used with `ragged2e'.\MessageBreak
4663         In this case, disable the `spacing' option}%
4664     }%
4665     \MT@check@active@set{sp}%
4666   \else
4667     \let\MT@spacing\relax
4668     \MT@info@nl{No adjustment of interword spacing}%
4669   \fi
4670 }

```

`\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

```

4671   \def\MT@setup@spacing@check{%
4672     \ifMT@spacing
4673       \ifMT@babel \else
4674         \ifnum\sfscode`. > 1500
4675           \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
4676             \MT@warning@nl{%
4677               \string\nonfrenchspacing\space is active. Adjustment of\MessageBreak
4678               interword spacing will disable it. You might want\MessageBreak
4679               to add `\

```

`\MT@setup@kerning`

```

4686   \def\MT@setup@kerning{%
4687     \ifMT@kerning
4688       \edef\MT@active@features{\MT@active@features,kn}%
4689       \pdfprependkern\@ne
4690       \pdfappendkern\@ne
4691       \MT@info@nl{Adjustment of character kerning enabled}%
4692       \MT@check@active@set{kn}%
4693     \else
4694       \let\MT@kerning\relax

```

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

```

4695     \MT@info@nl{No adjustment of character kerning}%
4696     \fi
4697   }
4698 </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.
4699 <pdfTeX-def|luaTeX-def>{}
4700 <*luaTeX-def>
4701   \def\MT@setup@tracking{%
4702     \ifMT@tracking
4703       \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
4704         or newer. Switching it off}{Upgrade luatex.}%
4705       \MT@trackingfalse
4706       \MT@let@nc{MT@tracking}\relax
4707     \else
4708       \MT@info@nl{No adjustment of tracking (luatex too old)}%
4709     \fi
4710   }
4711 }
4712 </luaTeX-def>
4713 <*pdfTeX-def|xetex-def|luaTeX-def>
4714   \def\MT@error@doesnt@work#1{%
4715     \csname ifMT@#1\endcsname
4716     \MT@error{The #1 feature only works with pdfTeX 1.40\MessageBreak
4717       or newer. Switching it off}
4718     {Upgrade pdfTeX.}%
4719     {Use pdfTeX instead.}%
4720     \csname MT@#1false\endcsname
4721     \MT@let@nc{MT@#1}\relax
4722   \else
4723     \MT@info@nl{No adjustment of #1%
4724 <pdfTeX-def>     \space(pdfTeX too old)%
4725     }%
4726   \fi
4727 }
4728 <pdfTeX-def|xetex-def> \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
4729 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
4730 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
4731 <pdfTeX-def>
4732 </pdfTeX-def|xetex-def|luaTeX-def>

\MT@setup@warntracking
4733 <letterspace>\MT@addto@setup
4734 <pdfTeX-def|luaTeX-def>\def\MT@setup@warntracking

\MT@warn@tracking@DVI   With pdfTeX, we issue a warning, when letterspacing in DVI mode, since it will
                        probably not work. We also switch on protrusion if it isn't already, to compensate
                        for the letterspacing kerns.
4735 <*pdfTeX-def|luaTeX-def|letterspace>
4736 {%
4737 <*pdfTeX-def|letterspace>
4738   \ifnum\pdfoutput<\@one
4739     \def\MT@warn@tracking@DVI{%
4740 <letterspace>     \MT@pdf@or@lua{%
4741       \MT@warning@nl{%
4742         You are using tracking/letterspacing in DVI mode.\MessageBreak
4743         This will probably not work, unless the post-\MessageBreak
4744         processing program (dvips, dviPfm(x), ...) is\MessageBreak
4745         able to create the virtual fonts on the fly}%
4746 <letterspace>     }\relax
4747     \MT@gl@t\MT@warn@tracking@DVI\relax
4748   }%
4749   \else

```

```

4750 </pdfTeX-def|letterspace>
4751   \def\MT@warn@tracking@DVI{%
4752     \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
4753     \MT@gl@et\MT@warn@tracking@DVI\relax
4754   }%
4755 </pdfTeX-def|letterspace> \fi
4756   \ifnum\MT@letterspace=\m@ne
4757     \let\MT@letterspace\MT@letterspace@default
4758   \else
4759     \MT@ls@too@large\MT@letterspace
4760   \fi
4761 }
4762 </pdfTeX-def|luatex-def|letterspace>
4763 <xetex-def>\let\MT@setup@warn@tracking\relax

```

`\MT@setup@noligatures` `\DisableLigatures` is only admissible in the preamble, therefore we can now disable the corresponding macro, if it was never called.

```

4764 <*pdfTeX-def|luatex-def>
4765 \def\MT@setup@noligatures{%
4766 <pdfTeX-def> \MT@requires@pdfTeX5{%
4767   \ifMT@noligatures \else
4768     \let\MT@noligatures\relax
4769   \fi
4770 <pdfTeX-def> }\relax
4771 }
4772 </pdfTeX-def|luatex-def>
4773 <xetex-def>\let\MT@setup@noligatures\relax

```

Remove the leading comma in `\MT@active@features`, and set the document switch to true.

```

4774 <*package>
4775 \MT@addto@setup{%
4776   \ifx\MT@active@features\@empty \else
4777     \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
4778   \fi
4779   \MT@documenttrue
4780 }

```

`\MT@set@babel@context` Interaction with `babel`.

```

4781 \def\MT@set@babel@context#1{%
4782   \MT@ifdefined@n@TF{MT@babel@#1}{%
4783     \MT@info{*** Changing to language context `#1'\MessageBreak\on@line}%
4784     \expandafter\MT@exp@one@n\expandafter\microtypecontext
4785     \csname MT@babel@#1\endcsname
4786   }{%
4787     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
4788   }%
4789 }

```

`\MT@shorthandoff` Active characters can only be switched off if `babel` isn't loaded after `microtype`.

```

4790 \@ifpackageloaded{babel}{
4791   \def\MT@shorthandoff#1#2{%
4792     \MT@info@n1{Switching off #1 babel's active characters (#2)}%
4793     \shorthandoff{#2}}
4794 }{
4795   \def\MT@shorthandoff#1#2{%
4796     \MT@error{You must load `babel' before `~\MT@MT'}
4797     {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
4798     active characters.}}
4799 }

```

We patch the language switching commands to enable language-dependent setup.

```

4800 \MT@addto@setup{%
4801   \ifMT@babel

```

```

4802 \ifpackageloaded{babel}{%
4803 \MT@info@n1{Redefining babel's language switching commands}%
4804 \let\MT@orig@select@language\select@language
4805 \def\select@language#1{%
4806 \MT@orig@select@language{#1}%
4807 \MT@set@babel@context{#1}%
4808 }%
4809 \let\MT@orig@foreign@language\foreign@language
4810 \def\foreign@language#1{%
4811 \MT@orig@foreign@language{#1}%
4812 \MT@set@babel@context{#1}%
4813 }%
4814 \ifMT@kerning

```

Disable French babel's active characters.

```

4815 \MT@if@false
4816 \MT@with@babel@and@T{french} \MT@if@true
4817 \MT@with@babel@and@T{frenchb} \MT@if@true
4818 \MT@with@babel@and@T{français} \MT@if@true
4819 \MT@with@babel@and@T{canadien} \MT@if@true
4820 \MT@with@babel@and@T{acadian} \MT@if@true
4821 \ifMT@if@MT@shorthandoff{French}{;!?}\fi

```

Disable Turkish babel's active characters.

```

4822 \MT@if@false
4823 \MT@with@babel@and@T{turkish} \MT@if@true
4824 \ifMT@if@MT@shorthandoff{Turkish}{:!=}\fi
4825 \fi

```

In case babel was loaded before microtype:

```

4826 \MT@set@babel@context\languagename
4827 }{%
4828 \MT@warning@n1{You did not load the babel package.\MessageBreak
4829 The `babel' option won't have any effect}%
4830 }%
4831 \fi
4832 }

```

Now we close the \fi from \ifMT@draft.

```

4833 \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.

```

4834 \selectfont}

```

\MT@curr@file This is the current file (hopefully with the correct extension).

```

4835 \edef\MT@curr@file{\jobname.tex}
4836 </package>

```

Finally, execute the setup macro at the end of the preamble, and empty it (the combine class calls it repeatedly).

```

4837 <*package|letterspace>
4838 <plain>\MT@requires@l@tex1{
4839 \AtBeginDocument{\MT@setup@ \MT@glet\MT@setup@\@empty}
4840 <plain>}\relax
4841 </package|letterspace>

```

Must come at the very, very end.

```

4842 <package>\MT@ifdefined@c@T\MT@setup@spacing@check
4843 <package> {\AtBeginDocument{\MT@setup@spacing@check}}

```

Restore catcodes.

```

4844 <package|letterspace>\MT@restore@catcodes

```

That was that.

15 Configuration files

Let's now write the font configuration files.

```
4845 (*config)
4846
```

15.1 Font sets

We first declare some sets in the main configuration file.

```
4847 (*m-t)
4848 %%% -----
4849 %%% FONT SETS
4850
4851 \DeclareMicrotypeSet{all}
4852   { }
4853
4854 \DeclareMicrotypeSet{allmath}
4855   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
4856
4857 \DeclareMicrotypeSet{alltext}
4858   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
4859
4860 \DeclareMicrotypeSet{allmath-nott}
4861   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
4862     family   = {rm*,sf*}
4863   }
4864
4865 \DeclareMicrotypeSet{alltext-nott}
4866   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4867     family   = {rm*,sf*}
4868   }
4869
4870 \DeclareMicrotypeSet{basicmath}
4871   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
4872     family   = {rm*,sf*},
4873     series   = {md*},
4874     size     = {normalsize,footnotesize,small,large}
4875   }
4876
4877 \DeclareMicrotypeSet{basictext}
4878   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
4879     family   = {rm*,sf*},
4880     series   = {md*},
4881     size     = {normalsize,footnotesize,small,large}
4882   }
4883
4884 \DeclareMicrotypeSet{smallcaps}
4885   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4886     shape    = {sc*,si,scit}
4887   }
4888
4889 \DeclareMicrotypeSet{footnotesize}
4890   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4891     size     = {-small}
4892   }
4893
4894 \DeclareMicrotypeSet{scriptsize}
4895   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
```

```

4896     size      = {-footnotesize}
4897   }
4898
4899 \DeclareMicrotypeSet{normal font}
4900   { font = */*/*/*/* }
4901

```

The default sets.

```

4902 %%% -----
4903 %%% DEFAULT SETS
4904
4905 \DeclareMicrotypeSetDefault[protrusion]{alltext}
4906 \DeclareMicrotypeSetDefault[expansion]{alltext-nott}
4907 \DeclareMicrotypeSetDefault[spacing]{alltext-nott}
4908 \DeclareMicrotypeSetDefault[kerning]{alltext}
4909 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
4910

```

15.2 Font variants and aliases

These are the variants I happen to be using (expert encoding, oldstyle numerals, swashes, alternative, display, inferior and superior numerals):

```

4911 %%% -----
4912 %%% FONT VARIANTS AND ALIASES
4913
4914 \DeclareMicrotypeVariants{x,j,w,a,d,0,1}

```

Other candidates: 2 (proportional digits), e (engraved), f (Fraktur), g (small text), h (shadow), l (outline), n (informal), p (ornaments), r (roman), s (sans serif), t (typewriter). I've omitted them since they seem hardly be used and/or they are actually more than just a variant, i.e., they shouldn't share a file.

Fonts that are 'the same': The fontspec package will set lmr as the default font, whose declarations for EU1/EU2/TU encoding are in mt-LatinModernRoman.cfg. Since 2016/12/03, the default encoding with XeTeX and LuaTeX in the L^AT_EX format is TU, even if fontspec is not loaded.

```

4915
4916 \MT@if@false
4917 \ifx\UnicodeEncodingName\undefined\else
4918   \MT@if@fstreq{\encodingdefault}{\UnicodeEncodingName}\MT@if@true\relax
4919 \fi
4920 \ifMT@fontspec\MT@if@true\fi
4921 \ifMT@if@
4922 % -- Computer/Latin Modern Roman
4923 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
4924   \else
4925 \DeclareMicrotypeAlias{lmr}{cmr}           % lmodern
4926 \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.

```

4927 \DeclareMicrotypeAlias{lmsy}{cmsy}       % ""
4928 \DeclareMicrotypeAlias{lmm}{cmm}        % ""
4929 \DeclareMicrotypeAlias{aer}{cmr}        % ae
4930 \DeclareMicrotypeAlias{zer}{cmr}        % zefonts
4931 \DeclareMicrotypeAlias{cmor}{cmr}       % eco
4932 \DeclareMicrotypeAlias{hfor}{cmr}       % hfoldsty

```

Another, new Computer Modern extension.

```
4933 \DeclareMicrotypeAlias{New Computer Modern}{Latin Modern Roman}
```

The packages pxfonts and txfonts fonts inherit Palatino and Times settings respectively, also the T_EX Gyre fonts Pagella and Termes (formerly: qfonts).

```
4934 %% -- Palatino
```

```
4935 \DeclareMicrotypeAlias{pxr} {ppl}          % pxfonts
```

```
4936 \DeclareMicrotypeAlias{qpl} {ppl}          % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)
```

The ‘FPL Neu’ fonts, a ‘re-implementation’ of Palatino.

```
4937 \DeclareMicrotypeAlias{fp9x}{pplx}        % FPL Neu
```

```
4938 \DeclareMicrotypeAlias{fp9j}{pplj}        % "
```

The newpx package, a replacement for pxfonts.

```
4939 \DeclareMicrotypeAlias{zpllf}{pplx}        % newpxtext
```

```
4940 \DeclareMicrotypeAlias{zplorf}{pplj}        % "
```

```
4941 \DeclareMicrotypeAlias{zpltlf}{pplx}        % "
```

```
4942 \DeclareMicrotypeAlias{zpltosf}{pplj}        % "
```

The domitian package.

```
4943 \DeclareMicrotypeAlias{Domitian-TLF} {pplx}% domitian
```

```
4944 \DeclareMicrotypeAlias{Domitian-TOfS}{pplj}% "
```

The OpenType versions:

```
4945 \DeclareMicrotypeAlias{Domitian}           {Palatino Linotype}
```

```
4946 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino Linotype}
```

```
4947 \DeclareMicrotypeAlias{Palatino LT Std} {Palatino Linotype}
```

```
4948 \DeclareMicrotypeAlias{Palatino}          {Palatino Linotype}
```

```
4949 \DeclareMicrotypeAlias{Asana Math}        {Palatino Linotype}
```

```
4950 %% -- Times New Roman
```

```
4951 \DeclareMicrotypeAlias{txr}{ptm}          % txfonts
```

The newtx package, a replacement for txfonts.

```
4952 \DeclareMicrotypeAlias{ntxlf}{ptmx}        % newtxtext
```

```
4953 \DeclareMicrotypeAlias{ntxosf}{ptmj}        % "
```

```
4954 \DeclareMicrotypeAlias{ntxtlf}{ptmx}        % "
```

```
4955 \DeclareMicrotypeAlias{ntxtosf}{ptmj}        % "
```

The tempora package.

```
4956 \DeclareMicrotypeAlias{Tempora-TLF}{ptmx} % tempora
```

```
4957 \DeclareMicrotypeAlias{Tempora-TOfS}{ptmj} % "
```

```
4958 \DeclareMicrotypeAlias{qtm}{ptm}          % TeX Gyre Termes (formerly: qfonts/QuasiTimes)
```

The step package.

```
4959 \DeclareMicrotypeAlias{STEP-TLF}{ptmx}     % step
```

```
4960 \DeclareMicrotypeAlias{STEP-TOfS}{ptmj}     % "
```

The stix and stix2 packages (the latter has departed a bit from being a Times clone, but still seems close enough).

```
4961 \DeclareMicrotypeAlias{stix} {ptm}         % stix
```

```
4962 \DeclareMicrotypeAlias{stix2}{ptm}         % stix2
```

More Times variants, to be checked: pns, mns (TimesNewRomanPS); mnt (TimesNewRomanMT, TimesNRSevenMT), mtm (TimesSmallTextMT); pte (TimesEuropa); ppt (TimesTen); TimesEighteen; TimesModernEF.

MicroPress’s Charter version (chmath).

```
4963 %% -- Charter
```

```
4964 \DeclareMicrotypeAlias{chr}{bch}           % CH Math
```

The XCharter package extends the Charter fonts.

```
4965 \DeclareMicrotypeAlias{XCharter-TLF} {bch} % XCharter
```

```
4966 \DeclareMicrotypeAlias{XCharter-TOfS}{bch} % "
```


The `mathdesign` package provides math fonts matching Bitstream Charter and URW Garamond.

```
4967 \DeclareMicrotypeAlias{mdbch}{bch}      % mathdesign/Charter
4968 %% -- Garamond
4969 \DeclareMicrotypeAlias{mdugm}{ugm}      % mathdesign/URW Garamond
```

The `garamondx` package, an extension of URW Garamond, providing small caps and oldstyle figures.

```
4970 \DeclareMicrotypeAlias{zgmX}{ugm}      % garamondx
4971 \DeclareMicrotypeAlias{zgmj}{ugm}      % "
4972 \DeclareMicrotypeAlias{zgmI}{ugm}      % "
4973 \DeclareMicrotypeAlias{zgmq}{ugm}      % "
4974 %% --
```

URW Letter Gothic is similar enough to Bitstream Letter Gothic to share the configuration.

```
4975 \DeclareMicrotypeAlias{ulg}{blg}      % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

The `eulervm` package virtually extends the Euler fonts.

```
4976 \DeclareMicrotypeAlias{zeur}{eur}      % Euler VM
4977 \DeclareMicrotypeAlias{zeus}{eus}      % "
```

Euro symbol fonts, to save some files.

```
4978 \DeclareMicrotypeAlias{zpeus}{zpeu}    % Adobe Euro sans -> serif
4979 \DeclareMicrotypeAlias{eurosans}{zpeu}  % Adobe Euro sans -> serif
4980 \DeclareMicrotypeAlias{euroitcs}{euroitc} % ITC Euro sans -> serif
4981
```

15.3 Interaction with babel

Contexts that are to be set when switching to a language.

```
4982 %%% -----
4983 %%% INTERACTION WITH THE `babel' PACKAGE
4984
4985 \DeclareMicrotypeBabelHook
4986   {english,UKenglish,british,USenglish,american}
4987   {kerning=, spacing=nonfrench}
4988
4989 \DeclareMicrotypeBabelHook
4990   {french,français,acadian,canadien}
4991   {kerning=french, spacing=}
4992
4993 \DeclareMicrotypeBabelHook
4994   {turkish}
4995   {kerning=turkish, spacing=}
4996
```

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.

With $X\TeX$ or $\text{Lua}\TeX$, in contrast, it is advisable to use the proper Unicode characters, or the font-specific glyph names prefixed with `'/` (cf. section 16).

15.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not $\text{C}\mathring{\text{E}}$ for O .

```
4997 </m-t>
4998 <*m-t|zpeu|mys>
4999 %%% -----
5000 %%% CHARACTER INHERITANCE
5001
5002 </m-t|zpeu|mys>
5003 <*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 , $\text{C}\mathring{\text{E}}$, œ .

```
5004 \DeclareCharacterInheritance
5005   { encoding = OT1 }
5006   { f = {011}, % ff
5007     i = {\i},
5008     j = {\j},
5009     O = {\O},
5010     o = {\o}
5011   }
5012
```

15.5.2 T1

Candidates here: 028 ('fi'), 029 ('fl'), 030 ('ffi'), 031 ('ffl'), 156 ('IJ' ligature, since $\text{L}\text{A}\text{T}\text{E}\text{X}$ 2005/12/01 accessible as `\IJ`), 188 ('ij', `\i j`), Æ , æ , $\text{C}\mathring{\text{E}}$, œ .

```
5013 \DeclareCharacterInheritance
5014   { encoding = T1 }
5015   { A = {\^A, \A, \^A, \-A, \A, \r A, \k A, \u A},
5016     a = {\^a, \a, \^a, \-a, \a, \r a, \k a, \u a},
5017     C = {\'C, \c C, \v C},
5018     c = {\'c, \c c, \v c},
5019     D = {\v D, \DH},
5020     d = {\v d, \dj},
5021     E = {\^E, \E, \^E, \E, \k E, \v E},
5022     e = {\^e, \e, \^e, \e, \k e, \v e},
5023     f = {027}, % ff
```

```

5024 G = {\u G},
5025 g = {\u g},
5026 I = {\^I, \'I, \^I, \^I, \.I},
5027 i = {\^i, \'i, \^i, \^i, \.i},
5028 j = {\j},
5029 L = {\L, \'L, \v L},
5030 l = {\l, \'l, \v l},
5031 N = {\^N, \-N, \v N},
5032 n = {\^n, \-n, \v n},
5033 O = {\O, \^O, \'O, \^O, \-O, \.O, \H O},
5034 o = {\o, \^o, \'o, \^o, \-o, \.o, \H o},
5035 R = {\'R, \v R},
5036 r = {\'r, \v r},
5037 S = {\'S, \c S, \v S, \SS},
5038 s = {\'s, \c s, \v s},
5039 T = {\c T, \v T},
5040 t = {\c t, \v t},
5041 U = {\^U, \'U, \^U, \^U, \H U, \r U},
5042 u = {\^u, \'u, \^u, \^u, \H u, \r u},
5043 Y = {\'Y, \.Y},
5044 y = {\'y, \.y},
5045 Z = {\'Z, \.Z, \v Z},
5046 z = {\'z, \.z, \v z}

```

The ‘soft hyphen’ often has reduced right side bearing so that it may already be protruded, hence no inheritance.

```

5047 % - = {127},
5048 }
5049

```

15.5.3 LY1

More characters: 008 (‘fl’), 012 (‘fi’), 014 (‘ffi’), 015 (‘ffl’), Æ, æ, Œ, œ.

```

5050 \DeclareCharacterInheritance
5051 { encoding = LY1 }
5052 { A = {\^A, \'A, \^A, \-A, \.A, \r A},
5053   a = {\^a, \'a, \^a, \-a, \.a, \r a},
5054   C = {\c C},
5055   c = {\c c},
5056   D = {\DH},
5057   E = {\^E, \'E, \^E, \.E},
5058   e = {\^e, \'e, \^e, \.e},
5059   f = {011}, % ff
5060   I = {\^I, \'I, \^I, \.I},
5061   i = {\^i, \'i, \^i, \.i, \.i},
5062   L = {\L},
5063   l = {\l},
5064   N = {\-N},
5065   n = {\-n},
5066   O = {\^O, \'O, \^O, \-O, \.O, \O},
5067   o = {\^o, \'o, \^o, \-o, \.o, \o},
5068   S = {\v S},
5069   s = {\v s},
5070   U = {\^U, \'U, \^U, \.U},
5071   u = {\^u, \'u, \^u, \.u},
5072   Y = {\'Y, \.Y},
5073   y = {\'y, \.y},
5074   Z = {\v Z},
5075   z = {\v z}
5076 }
5077

```

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.

```

5078 \DeclareCharacterInheritance
5079   { encoding = OT4 }
5080   { A = {\k A},
5081     a = {\k a},
5082     C = {\'C},
5083     c = {\'c},
5084     E = {\k E},
5085     e = {\k e},
5086     f = {011}, % ff
5087     i = {\i},
5088     j = {\j},
5089     L = {\L},
5090     l = {\l},
5091     N = {\'N},
5092     n = {\'n},
5093     O = {\0,\'0},
5094     o = {\0,\'o},
5095     S = {\'S},
5096     s = {\'s},
5097     Z = {\'Z,\.Z},
5098     z = {\'z,\.z},
5099     \textquotedblleft = "FF
5100   }
5101

```

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.

```

5102 \DeclareCharacterInheritance
5103   { encoding = QX }
5104   { A = {\^A,\'A,\^A,\^A,\-A,\"A,\k A,\AA},
5105     a = {\^a,\'a,\^a,\^a,\-a,\"a,\k a,\aa},
5106     C = {\'C,\c C},
5107     c = {\'c,\c c},
5108     D = {\DH},
5109     E = {\^E,\'E,\^E,\^E,\"E,\k E},
5110     e = {\^e,\'e,\^e,\^e,\"e,\k e},
5111     f = {011}, % ff
5112     I = {\^I,\'I,\^I,\^I,\"I,\k I},
5113     i = {\^i,\'i,\^i,\^i,\"i,\k i,\i},
5114     j = {\j},
5115     L = {\L},
5116     l = {\l},
5117     N = {\'N,\-N},
5118     n = {\'n,\-n},
5119     O = {\0,\^0,\'0,\^0,\-0,\"0},
5120     o = {\0,\^o,\'o,\^o,\-o,\"o},

```

The Romanian `\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.

```

5121   S = {\'S,\c S,\textcommabelow S,\v S},
5122   s = {\'s,\c s,\textcommabelow s,\v s},
5123   T = {\c T,\textcommabelow T},

```

¹⁷ Contributed by Maciej Eder.

¹⁸ Cf. <http://tug.org/pipermail/tex-live/2008-August/017204.html>

```

5124 t = {\c t,\textcommabelow t},
5125 U = {\^U,\'U,\^U,\^U,\k U},
5126 u = {\^u,\'u,\^u,\^u,\k u},
5127 Y = {\'Y,\"Y},
5128 y = {\'y,\"y},
5129 Z = {\'Z,\.Z,\v Z},
5130 z = {\'z,\.z,\v z},
5131 . = \textellipsis
5132 }
5133

```

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.

```

5134 \DeclareCharacterInheritance
5135 { encoding = T5 }
5136 { A = {\^A,\'A,\-A,\h A,\d A,\^A,\u A,
5137       \^Acircumflex,\'Acircumflex,\-Acircumflex,\hAcircumflex,\dAcircumflex,
5138       \^Abreve,\'Abreve,\-Abreve,\hAbreve,\dAbreve},
5139 a = {\^a,\'a,\-a,\h a,\d a,\^a,\u a,
5140       \^acircumflex,\'acircumflex,\-acircumflex,\hacircumflex,\dacircumflex,
5141       \^abreve,\'abreve,\-abreve,\h\abreve,\d\abreve},
5142 D = {\DJ},
5143 d = {dj},
5144 E = {\^E,\'E,\-E,\h E,\d E,\^E,
5145       \^Ecircumflex,\'Ecircumflex,\-Ecircumflex,\hEcircumflex,\dEcircumflex},
5146 e = {\^e,\'e,\-e,\h e,\d e,\^e,
5147       \^ecircumflex,\'ecircumflex,\-ecircumflex,\h\ecircumflex,\d\ecircumflex},
5148 I = {\^I,\'I,\-I,\h I,\d I},
5149 i = {\^i,\'i,\-i,\h i,\d i,\i},
5150 O = {\^O,\'O,\-O,\h O,\d O,\^O,\horn O,
5151       \^Ocircumflex,\'Ocircumflex,\-Ocircumflex,\hOcircumflex,\dOcircumflex,
5152       \^Ohorn,\'Ohorn,\-Ohorn,\hOhorn,\dOhorn},
5153 o = {\^o,\'o,\-o,\h o,\d o,\^o,\horn o,
5154       \^ocircumflex,\'ocircumflex,\-ocircumflex,\h\ocircumflex,\d\ocircumflex,
5155       \^ohorn,\'ohorn,\-ohorn,\h\ohorn,\d\ohorn},
5156 U = {\^U,\'U,\-U,\h U,\d U,\horn U,
5157       \^Uhorn,\'Uhorn,\-Uhorn,\hUhorn,\dUhorn},
5158 u = {\^u,\'u,\-u,\h u,\d u,\horn u,
5159       \^uhorn,\'uhorn,\-uhorn,\h\uhorn,\d\uhorn},
5160 Y = {\^Y,\'Y,\-Y,\h Y,\d Y},
5161 y = {\^y,\'y,\-y,\h y,\d y}
5162 }
5163

```

15.5.7 EU1, EU2, TU

The EU1 (X_YTeX), EU2 (LuaTeX), and, since fontspec version 2.5, TU encodings are not well-defined in the sense that they don't contain a fixed number of glyphs, all of which must be present. OpenType fonts may contain thousands of glyphs, but we only define those that should be present in every font (basically T1). This inheritance list should be overridden by font-specific ones.

```

5164 \DeclareCharacterInheritance
5165 { encoding = {EU1,EU2,TU} }
5166 { A = {\^A,\'A,\^A,\-A,\"A,\r A,\k A,\u A},
5167 a = {\^a,\'a,\^a,\-a,\"a,\r a,\k a,\u a},
5168 C = {\'C,\c C,\v C},
5169 c = {\'c,\c c,\v c},
5170 D = {\v D,\DH},
5171 d = {\v d,\dj},

```

```

5172 E = {\^E,\'E,\^E,\^E,\k E,\v E},
5173 e = {\^e,\'e,\^e,\^e,\k e,\v e},
5174 % f = {/f_f}, % sometimes /f_f, sometimes /ff
5175 G = {\u G},
5176 g = {\u g},
5177 I = {\^I,\'I,\^I,\^I,\.I},
5178 i = {\^i,\'i,\^i,\^i,\i},
5179 % j = {\j},
5180 L = {\L,\'L,\v L},
5181 l = {\l,\'l,\v l},
5182 N = {\^N,\'N,\v N},
5183 n = {\'n,\-n,\v n},
5184 O = {\O,\^O,\'O,\^O,\-O,\"O,\H O},
5185 o = {\o,\^o,\'o,\^o,\-o,\"o,\H o},
5186 R = {\'R,\v R},
5187 r = {\'r,\v r},
5188 S = {\'S,\c S,\v S}, % \SS
5189 s = {\'s,\c s,\v s},
5190 T = {\c T,\v T},
5191 t = {\c t,\v t},
5192 U = {\^U,\'U,\^U,\"U,\H U,\r U},
5193 u = {\^u,\'u,\^u,\"u,\H u,\r u},
5194 Y = {\'Y,\"Y},
5195 y = {\'y,\"y},
5196 Z = {\'Z,\.Z,\v Z},
5197 z = {\'z,\.z,\v z}
5198 }
5199
5200 </m-t>

```

15.5.8 Euro symbols

Make Euro symbols settings simpler.

```

5201 < *zpeu >
5202 \DeclareCharacterInheritance
5203 { encoding = U,
5204   family   = {zpeu,zpeus,eurosans} }
5205 { E = 128 }
5206
5207 < /zpeu >
5208 < *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.

```

5209 \DeclareCharacterInheritance
5210 { encoding = {OT1,U},
5211   family   = mvs }
5212 { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
5213
5214 < /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.

```

5215 < *m-t >
5216 %%% -----
5217 %%% TRACKING/LETTERSPACING
5218
5219 \SetTracking
5220 [ name          = default,

```

```

5221 no ligatures = {f} ]
5222 { encoding = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
5223 { }
5224

```

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).

```

5225 %%% -----
5226 %%% EXPANSION
5227
5228 \SetExpansion
5229 [ name = default ]
5230 { encoding = {OT1,OT4,QX,T1,LY1} }
5231 {
5232 A = 500, a = 700,
5233 \AE = 500, \ae = 700,
5234 B = 700, b = 700,
5235 C = 700, c = 700,
5236 D = 500, d = 700,
5237 E = 700, e = 700,
5238 F = 700,
5239 G = 500, g = 700,
5240 H = 700, h = 700,
5241 K = 700, k = 700,
5242 M = 700, m = 700,
5243 N = 700, n = 700,
5244 O = 500, o = 700,
5245 \OE = 500, \oe = 700,
5246 P = 700, p = 700,
5247 Q = 500, q = 700,
5248 R = 700,
5249 S = 700, s = 700,
5250 U = 700, u = 700,
5251 W = 700, w = 700,
5252 Z = 700, z = 700,
5253 2 = 700,
5254 3 = 700,
5255 6 = 700,
5256 8 = 700,
5257 9 = 700
5258 }
5259

```

Settings for Cyrillic T2A encoding.¹⁹

```

5260 \SetExpansion
5261 [ name = T2A ]
5262 { encoding = T2A }
5263 {
5264 A = 500, a = 700,
5265 B = 700, b = 700,
5266 C = 700, c = 700,
5267 D = 500, d = 700,
5268 E = 700, e = 700,
5269 F = 700,
5270 G = 500, g = 700,
5271 H = 700, h = 700,
5272 K = 700, k = 700,
5273 M = 700, m = 700,
5274 N = 700, n = 700,
5275 O = 500, o = 700,

```

19 Contributed by *Karl Karlsson*.

```

5276     P = 700,      p = 700,
5277     Q = 500,      q = 700,
5278     R = 700,
5279     S = 700,      s = 700,
5280     U = 700,      u = 700,
5281     W = 700,      w = 700,
5282     Z = 700,      z = 700,
5283     2 = 700,
5284     3 = 700,
5285     6 = 700,
5286     8 = 700,
5287     9 = 700,
5288     \CYRA = 500,   \cyra = 700,
5289     \CYRB = 700,   \cyrb = 700,
5290     \CYRV = 700,   \cyrv = 700,
5291     \CYRG = 700,   \cyrg = 700,
5292     \CYRD = 700,   \cyrd = 700,
5293     \CYRE = 700,   \cyre = 700,
5294     \CYRZH = 700,  \cyrzh = 700,
5295     \CYRZ = 700,   \cyrz = 700,
5296     \CYRI = 700,   \cyri = 700,
5297     \CYRISHRT = 700, \cyrishrt = 700,
5298     \CYRK = 700,   \cyrk = 700,
5299     \CYRL = 700,   \cyrl = 700,
5300     \CYRM = 700,   \cyrm = 700,
5301     \CYRN = 700,   \cyrn = 700,
5302     \CYRO = 500,   \cyro = 700,
5303     \CYRP = 700,   \cyrp = 700,
5304     \CYRR = 700,   \cyrr = 700,
5305     \CYRS = 700,   \cyrs = 700,
5306     \CYRT = 700,   \cyrt = 700,
5307     \CYRU = 700,   \cyru = 700,
5308     \CYRF = 700,   \cyrf = 700,
5309     \CYRH = 700,   \cyrh = 700,
5310     \CYRC = 700,   \cyrc = 700,
5311     \CYRCH = 700,  \cyrch = 700,
5312     \CYRSH = 700,  \cyrsh = 700,
5313     \CYRSHCH = 700, \cyrshch = 700,
5314     \CYRHRDSN = 700, \cyhrdsn = 700,
5315     \CYRERY = 700,  \cyrery = 700,
5316     \CYRSFTSN = 700, \cyrsftsn = 700,
5317     \CYREREV = 700,  \cyrerev = 700,
5318     \CYRYU = 700,   \cyryu = 700,
5319     \CYRYA = 700,   \cyrya = 700
5320   }
5321

```

T5 encoding does not contain \AE, \ae, \OE and \oe.

```

5322 \SetExpansion
5323   [ name      = T5 ]
5324   { encoding = T5 }
5325   {
5326     A = 500,      a = 700,
5327     B = 700,      b = 700,
5328     C = 700,      c = 700,
5329     D = 500,      d = 700,
5330     E = 700,      e = 700,
5331     F = 700,
5332     G = 500,      g = 700,
5333     H = 700,      h = 700,
5334     K = 700,      k = 700,
5335     M = 700,      m = 700,
5336     N = 700,      n = 700,
5337     O = 500,      o = 700,
5338     P = 700,      p = 700,

```



```

5339   Q = 500,   q = 700,
5340   R = 700,
5341   S = 700,   s = 700,
5342   U = 700,   u = 700,
5343   W = 700,   w = 700,
5344   Z = 700,   z = 700,
5345   2 = 700,
5346   3 = 700,
5347   6 = 700,
5348   8 = 700,
5349   9 = 700
5350   }
5351
5352 </m-t>

```

15.8 Character protrusion

```

5353 %%% -----
5354 %%% PROTRUSION
5355

```

For future historians, Hàn Thê Thành's original settings (from `protcode.tex`, converted to `microtype` notation).

```

\SetProtrusion
[ name      = thanh ]
{ encoding = OT1 }
{
  A = {50,50},
  F = { ,50},
  J = {50, },
  K = { ,50},
  L = { ,50},
  T = {50,50},
  V = {50,50},
  W = {50,50},
  X = {50,50},
  Y = {50,50},
  k = { ,50},
  r = { ,50},
  t = { ,50},
  v = {50,50},
  w = {50,50},
  x = {50,50},
  y = {50,50},
  . = { ,700},   {,}= { ,700},
  : = { ,500},   ; = { ,500},
  ! = { ,200},   ? = { ,200},
  ( = {50, },   ) = { ,50},
  - = { ,700},
  \textendash   = { ,300},   \textemdash       = { ,200},
  \textquoteleft = {700, },   \textquoteright   = { ,700},
  \textquotedblleft = {500, }, \textquotedblright = { ,500}
}

```

15.8.1 Normal

The default settings always use the most moderate value.

```

5356 <*cfg-t>
5357 \SetProtrusion
5358 <m-t> [ name      = default ]

```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

5359 *<bch>* [name = bch-default]

- Bitstream Letter Gothic (blg)

5360 *<blg>* [name = blg-default]

- Computer Modern Roman (cmr)

5361 *<cmr>* [name = cmr-default]

- Adobe Garamond (pad, padx, padj)

5362 *<pad>* [name = pad-default]

- Minion²⁰ (pmnx, pmnj)

5363 *<pmn>* [name = pmnj-default]

- Palatino (ppl, pplx, pplj)

5364 *<ppl>* [name = ppl-default]

- Times (ptm, ptmx, ptmj)

5365 *<ptm>* [name = ptm-default]

- URW Garamond (ugm)

5366 *<ugm>* [name = ugm-default]
 5367 *<m-t|cmr|pmn>* { }
 5368 *<bch|blg|pad|ugm>* { encoding = OT1,
 5369 *<ppl|ptm>* { encoding = {OT1,OT4},
 5370 *<bch>* family = bch }
 5371 *<blg>* family = blg }
 5372 *<pad>* family = {pad,padx,padj }
 5373 *<ppl>* family = {ppl,pplx,pplj }
 5374 *<ptm>* family = {ptm,ptmx,ptmj }
 5375 *<ugm>* family = ugm }
 5376 {
 5377 *<m-t|bch|blg|cmr|pad|pmn|ppl|ptm>* A = {50,50},
 5378 *<ugm>* A = {50,100},
 5379 *<pad|ptm>* \AE = {50, },
 5380 *<ugm>* \AE = {150,50},
 5381 *<ugm>* B = { ,50},
 5382 *<bch|pad|pmn|ugm>* C = {50, },
 5383 *<bch|pad|pmn>* D = { ,50},
 5384 *<ugm>* D = { ,70},
 5385 *<ugm>* E = { ,50},
 5386 *<m-t|bch|cmr|pad|pmn|ptm>* F = { ,50},
 5387 *<ugm>* F = { ,70},
 5388 *<bch|pad|pmn>* G = {50, },
 5389 *<ugm>* G = {50,50},
 5390 *<blg>* I = {150,150},
 5391 *<m-t|cmr|pad|pmn|ppl|ptm|ugm>* J = {50, },
 5392 *<bch|blg>* J = {100, },
 5393 *<!blg>* K = { ,50},
 5394 *<blg>* K = {50, },
 5395 *<m-t|bch|cmr|pad|pmn|ppl>* L = { ,50},
 5396 *<blg>* L = { ,150},
 5397 *<ptm>* L = { ,80},
 5398 *<ugm>* L = { ,120},
 5399 *<bch|pad|pmn|ugm>* O = {50,50},
 5400 *<pad>* \OE = {50, },

5401 $\langle ugm \rangle \quad \backslash OE = \{50,50\},$
5402 $\langle blg \rangle \quad P = \{ \quad ,100\},$
5403 $\langle ugm \rangle \quad P = \{ \quad ,50\},$
5404 $\langle bch|pad|pmn \rangle \quad Q = \{50,70\},$
5405 $\langle ugm \rangle \quad Q = \{50,50\},$
5406 $\langle bch \rangle \quad R = \{ \quad ,50\},$
5407 $\langle ugm \rangle \quad R = \{ \quad ,70\},$
5408 $\langle m-t|bch|cmr|pad|pmn|ppl|ptm \rangle \quad T = \{50,50\},$
5409 $\langle blg \rangle \quad T = \{100,100\},$
5410 $\langle ugm \rangle \quad T = \{70,70\},$
5411 $\langle m-t|bch|cmr|pad|pmn|ppl|ptm \rangle \quad V = \{50,50\},$
5412 $\langle blg|ugm \rangle \quad V = \{70,70\},$
5413 $\langle m-t|bch|cmr|pad|pmn|ppl|ptm \rangle \quad W = \{50,50\},$
5414 $\langle ugm \rangle \quad W = \{70,70\},$
5415 $\langle m-t|bch|cmr|pad|pmn|ppl|ptm \rangle \quad X = \{50,50\},$
5416 $\langle ugm \rangle \quad X = \{50,70\},$
5417 $\langle m-t|bch|cmr|pad|pmn|ppl \rangle \quad Y = \{50,50\},$
5418 $\langle blg|ptm|ugm \rangle \quad Y = \{80,80\},$
5419 $\langle ugm \rangle \quad Z = \{50,50\},$
5420 $\langle blg \rangle \quad f = \{150,100\},$
5421 $\langle blg \rangle \quad i = \{150,150\},$
5422 $\langle blg \rangle \quad j = \{100,100\},$
5423 $\langle m-t|bch|cmr|pad|pmn|ppl|ptm \rangle \quad k = \{ \quad ,50\},$
5424 $\langle ugm \rangle \quad k = \{ \quad ,70\},$
5425 $\langle blg \rangle \quad l = \{150,150\},$
5426 $\langle pmn \rangle \quad l = \{ \quad ,-50\},$
5427 $\langle pad|ppl \rangle \quad p = \{50,50\},$
5428 $\langle ugm \rangle \quad p = \{ \quad ,50\},$
5429 $\langle pad|ppl \rangle \quad q = \{50, \quad \},$
5430 $\langle !blg \rangle \quad r = \{ \quad ,50\},$
5431 $\langle blg \rangle \quad r = \{100, 80\},$
5432 $\langle cmr|pad|pmn \rangle \quad t = \{ \quad ,70\},$
5433 $\langle bch \rangle \quad t = \{ \quad ,50\},$
5434 $\langle blg \rangle \quad t = \{150, 80\},$
5435 $\langle ugm \rangle \quad t = \{ \quad ,100\},$
5436 $\langle m-t|bch|cmr|pad|pmn|ppl|ptm \rangle \quad v = \{50,50\},$
5437 $\langle blg \rangle \quad v = \{100,100\},$
5438 $\langle ugm \rangle \quad v = \{50,70\},$
5439 $\langle m-t|bch|cmr|pad|pmn|ppl|ptm \rangle \quad w = \{50,50\},$
5440 $\langle ugm \rangle \quad w = \{50,70\},$
5441 $\langle !blg \rangle \quad x = \{50,50\},$
5442 $\langle blg \rangle \quad x = \{100,100\},$
5443 $\langle m-t|bch|pad|pmn \rangle \quad y = \{ \quad ,50\},$
5444 $\langle blg \rangle \quad y = \{ 50,100\},$
5445 $\langle cmr|ppl|ptm \rangle \quad y = \{50,70\},$
5446 $\langle ugm \rangle \quad y = \{ \quad ,70\},$

5447 $\langle cmr \rangle \quad 0 = \{ \quad ,50\},$
5448 $\langle m-t \rangle \quad 1 = \{50,50\},$
5449 $\langle bch|blg|pad|ptm|ugm \rangle \quad 1 = \{150,150\},$
5450 $\langle cmr \rangle \quad 1 = \{100,200\},$
5451 $\langle pmn \rangle \quad 1 = \{ \quad ,50\},$
5452 $\langle ppl \rangle \quad 1 = \{100,100\},$
5453 $\langle bch|cmr|pad|ugm \rangle \quad 2 = \{50,50\},$
5454 $\langle blg \rangle \quad 2 = \{ \quad ,100\},$
5455 $\langle bch|pmn \rangle \quad 3 = \{50, \quad \},$
5456 $\langle cmr|pad|ugm \rangle \quad 3 = \{50,50\},$
5457 $\langle blg \rangle \quad 3 = \{100, \quad \},$
5458 $\langle m-t|pad \rangle \quad 4 = \{50,50\},$
5459 $\langle bch \rangle \quad 4 = \{100,50\},$
5460 $\langle blg \rangle \quad 4 = \{100, \quad \},$
5461 $\langle cmr|ugm \rangle \quad 4 = \{70,70\},$
5462 $\langle pmn \rangle \quad 4 = \{50, \quad \},$
5463 $\langle ptm \rangle \quad 4 = \{70, \quad \},$
5464 $\langle cmr \rangle \quad 5 = \{ \quad ,50\},$
5465 $\langle pad \rangle \quad 5 = \{50,50\},$

```

5466 <bch> 6 = {50, },
5467 <cmr> 6 = { ,50},
5468 <pad> 6 = {50,50},
5469 <m-t> 7 = {50,50},
5470 <bch|pad|pmn|ugm> 7 = {50,80},
5471 <blg> 7 = {100,100},
5472 <cmr|ptm> 7 = {50,100},
5473 <ppl> 7 = { ,50},
5474 <cmr> 8 = { ,50},
5475 <bch|pad> 9 = {50,50},
5476 <cmr> 9 = { ,50},
5477 <m-t|cmr|pad|pmn|ppl|ptm|ugm> . = { ,700},
5478 <bch> . = { ,600},
5479 <blg> . = {400,500},
5480 <!blg> {,}= { ,500},
5481 <blg> {,}= {300,400},
5482 <m-t|cmr|pad|pmn|ppl|ptm|ugm> : = { ,500},
5483 <bch> : = { ,400},
5484 <blg> : = {300,400},
5485 <m-t|bch|pad|pmn|ptm> ; = { ,300},
5486 <blg> ; = {200,300},
5487 <cmr|ppl> ; = { ,500},
5488 <ugm> ; = { ,400},
5489 <!blg> ! = { ,100},
5490 <blg> ! = {200,200},
5491 <m-t|pad|pmn|ptm> ? = { ,100},
5492 <bch|cmr|ppl|ugm> ? = { ,200},
5493 <blg> ? = {150,150},
5494 <pmn> " = {300,300},
5495 <m-t|bch|cmr|pad|pmn|ppl> @ = {50,50},
5496 <ptm> @ = {100,100},
5497 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm> - = {200,250},
5498 <ugm> - = {300,350},
5499 <pad|ppl|ptm> & = {50,100},
5500 <ugm> & = { ,100},
5501 <m-t|cmr|pad|pmn> \% = {50,50},
5502 <bch> \% = { ,50},
5503 <ppl|ptm> \% = {100,100},
5504 <ugm> \% = {50,100},
5505 <blg> \# = {100,100},
5506 <m-t|ppl|ptm|ugm> * = {200,200},
5507 <bch|pmn> * = {200,300},
5508 <blg> * = {150,200},
5509 <cmr|pad> * = {300,300},
5510 <m-t|cmr|ppl|ptm> + = {250,250},
5511 <bch> + = {150,250},
5512 <pad> + = {300,300},
5513 <blg|pmn> + = {150,200},
5514 <ugm> + = {250,300},
5515 <blg|ugm> {=} = {200,200},
5516 <m-t|pad|pmn|ptm> ( = {100, }, ) = { ,200},
5517 <bch|ugm> ( = {200, }, ) = { ,200},
5518 <cmr|blg> ( = {300, }, ) = { ,300},
5519 <ppl> ( = {100, }, ) = { ,300},
5520 <bch|pmn> [ = {100, }, ] = { ,100},
5521 <blg> [ = {300,100}, ] = { ,300},

5522 <m-t|pad|pmn|ptm> / = {100,200},
5523 <bch> / = { ,200},
5524 <blg> / = {300,300},
5525 <cmr|ppl> / = {200,300},
5526 <ugm> / = {100,300},
5527 <m-t|ptm> - = {500,500},
5528 <bch|cmr|ppl> - = {400,500},
5529 <blg> - = {300,400},
5530 <pad> - = {300,500},

```

```

5531 <pmn> - = {200,400},
5532 <ugm> - = {500,600},
5533 <blg> < = {200,100}, > = {100,200},
5534 <blg> - = {150,250},
5535 <blg> | = {250,250},
5536 <m-t|pmn> \textendash = {200,200}, \textemdash = {150,150},
5537 <bch> \textendash = {200,300}, \textemdash = {150,250},
5538 <cmr> \textendash = {400,300}, \textemdash = {300,200},
5539 <pad|ppl|ptm> \textendash = {300,300}, \textemdash = {200,200},
5540 <ugm> \textendash = {250,300}, \textemdash = {250,250},

```

Why settings for left *and* right quotes? Because in some languages they might be used like that (see the csquotes package for examples).

```

5541 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
5542 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
5543 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
5544 <pad|ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
5545 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
5546 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
5547 <m-t|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
5548 <blg> \textquotedblright = {300,400}
5549 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
5550 <pad|ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5551 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
5552 }
5553

```

Greek uppercase letters are in OT1 encoding only.

```

5554 <*m-t|cmr|pmn>
5555 \SetProtrusion
5556 <m-t> [ name = OT1-default,
5557 <cmr> [ name = cmr-OT1,
5558 <pmn> [ name = pmnj-OT1,
5559 <m-t> load = default ]
5560 <cmr> load = cmr-default ]
5561 <pmn> load = pmnj-default ]
5562 <m-t> { encoding = OT1 }
5563 <cmr> { encoding = {OT1,OT4},
5564 <pmn> { encoding = OT1,
5565 <cmr> family = cmr }
5566 <pmn> family = pmnj }
5567 {
5568 <m-t|cmr> \AE = {50, },
5569 <pmn> \OE = {50, }
5570 <*cmr>
5571 "00 = { ,150}, % \Gamma
5572 "01 = {100,100}, % \Delta
5573 "02 = { 50, 50}, % \Theta
5574 "03 = {100,100}, % \Lambda
5575 "06 = { 50, 50}, % \Sigma
5576 "07 = {100,100}, % \Upsilon
5577 "08 = { 50, 50}, % \Phi
5578 "09 = { 50, 50} % \Psi

```

Remaining slots can be found in the source file.

```

5579 </cmr>
5580 }
5581
5582 </m-t|cmr|pmn>

```

T1 and LY1 encodings contain some more characters. The default list will be loaded first. For X_YT_EX (EU1) and LuaT_EX (EU2) we simply use the T1 list as default (for now).

```

5583 \SetProtrusion

```

```

5584 <m-t> [ name = T1-default,
5585 <bch> [ name = bch-T1,
5586 <blg> [ name = blg-T1,
5587 <cmr> [ name = cmr-T1,
5588 <pad> [ name = pad-T1,
5589 <pmn> [ name = pmnj-T1,
5590 <ppl> [ name = ppl-T1,
5591 <ptm> [ name = ptm-T1,
5592 <ugm> [ name = ugm-T1,
5593 <m-t> load = default ]
5594 <bch> load = bch-default ]
5595 <blg> load = blg-default ]
5596 <cmr> load = cmr-default ]
5597 <pad> load = pad-default ]
5598 <pmn> load = pmnj-default ]
5599 <ppl> load = ppl-default ]
5600 <ptm> load = ptm-default ]
5601 <ugm> load = ugm-default ]
5602 <m-t> { encoding = {T1,LY1,EU1,EU2,TU} }
5603 <bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
5604 <blg|ptm|ugm> { encoding = {T1},
5605 <bch> family = bch }
5606 <blg> family = blg }
5607 <cmr> family = cmr }
5608 <pad> family = {pad,padx,padj} }
5609 <pmn> family = pmnj }
5610 <ppl> family = {ppl,pplx,pplj} }
5611 <ptm> family = {ptm,ptmx,ptmj} }
5612 <ugm> family = ugm }
5613 {
5614 <m-t|cmr> \AE = {50, },
5615 <bch|pmn> \OE = {50, },
5616 <pmn> \TH = { ,50},
5617 <blg> \v L = { ,250},
5618 <blg> \v d = { ,250},
5619 <blg> \v l = { ,250},
5620 <blg> \v t = { ,250},
5621 <blg> 127 = {300,400},
5622 <blg> 156 = {100, }, % IJ
5623 <blg> 188 = { 80, 80}, % ij
5624 <m-t|bch|pad|pmn|ppl|ptm> _ = {100,100},
5625 <cmr> _ = {200,200},
5626 <ugm> _ = {100,200},
5627 <m-t|pad|pmn|ptm> \textbackslash = {100,200},
5628 <bch> \textbackslash = {150,200},
5629 <blg> \textbackslash = {250,300},
5630 <cmr|ppl> \textbackslash = {200,300},
5631 <ugm> \textbackslash = {100,300},
5632 <ugm> \textbar = {200,200},
5633 <blg> \textendash = {300,300}, \textemdash = {150,150},
5634 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
5635 <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.

```

5636 <m-t|cmr|pad|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5637 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
5638 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5639 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5640 <blg> \guilsinglleft = {300,500}, \guilsinglright = {300,500},
5641 <cmr|pad|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5642 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
5643 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5644 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5645 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},

```

```

5646 <blg|pad|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5647 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
5648 <m-t|bch|cmr|pad|pnn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
5649 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
5650 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
5651 <m-t|cmr|pad|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
5652 <bch|blg|pnn> \textbraceleft = {200, }, \textbraceright = { ,300},
5653 <m-t|bch|cmr|pad|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
5654 <pnn> \textless = {100, }, \textgreater = { ,100},
5655 <pnn> \textvisiblespace = {100,100} % not in LY1

5656 }
5657

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

5658 <*cmr>
5659 \SetProtrusion
5660 [ name = lmr-T1,
5661 load = cmr-T1 ]
5662 { encoding = {T1,LY1},
5663 family = lmr }
5664 {
5665 \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5666 }
5667
5668 </cmr>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).²¹

```

5669 <*m-t|cmr|pnn>
5670 \SetProtrusion
5671 <m-t> [ name = T2A-default,
5672 <cmr> [ name = cmr-T2A,
5673 <pnn> [ name = pnnj-T2A,
5674 <m-t> load = default ]
5675 <cmr> load = cmr-default ]
5676 <pnn> load = pnnj-default ]
5677 { encoding = T2A,
5678 <m-t> }
5679 <cmr> family = cmr }
5680 <pnn> family = pnnj }
5681 {
5682 \CYRA = {50,50},
5683 \CYRG = { ,50},
5684 \CYRK = { ,50},
5685 \CYRT = {50,50},
5686 \CYRH = {50,50},
5687 \CYRU = {50,50},
5688 <pnn> \CYRS = {50, },
5689 <pnn> \CYRO = {50,50},
5690 \cyrk = { ,50},
5691 \cyrg = { ,50},
5692 \cyrh = {50,50},
5693 <m-t|pnn> \cyru = {50,50},
5694 <cmr> \cyru = {50,70},
5695 <m-t> - = {100,100},
5696 <cmr> - = {200,200},
5697 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,400},
5698 <cmr> \textbackslash = {200,300}, \quotedblbase = {400,400},
5699 <pnn> \textbackslash = {100,200}, \quotedblbase = {300,300},
5700 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},
5701 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5702 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5703 <pnn> \guillemotleft = {200,200}, \guillemotright = {150,300},

```

21 Contributed by Karl Karlsson.

```

5704 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5705 <pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
5706 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
5707 <pmn> \textless = {100, }, \textgreater = { ,100}
5708 }
5709
5710 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).²² It also includes some glyphs otherwise in TS1.

```

5711 <*m-t|ptm>
5712 \SetProtrusion
5713 <m-t> [ name = QX-default,
5714 <ptm> [ name = ptm-QX,
5715 <m-t> load = default ]
5716 <ptm> load = ptm-default ]
5717 <m-t> { encoding = QX }
5718 <ptm> { encoding = QX,
5719 <ptm> family = {ptm,ptmx,ptmj} }
5720 {
5721 \AE = {50, },
5722 <ptm> * = {200,200},
5723 {=} = {100,100},
5724 \textunderscore = {100,100},
5725 \textbackslash = {100,200},
5726 \quotedblbase = {400,400},
5727 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5728 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5729 \textexclamdown = {100, }, \textquestiondown = {100, },
5730 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
5731 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
5732 \textless = {200,100}, \textgreater = {100,200},
5733 \textminus = {200,200}, \textdegree = {300,300},
5734 <m-t> \copyright = {100,100}, \textregistered = {100,100}
5735 <ptm> \copyright = {100,150}, \textregistered = {100,150},
5736 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
5737 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
5738 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
5739 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
5740 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
5741 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
5742 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
5743 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
5744 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
5745 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
5746 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
5747 <ptm> \textperthousand = { ,50}
5748 }
5749
5750 </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.

```

5751 <*cmr|bch>
5752 \SetProtrusion
5753 <cmr> [ name = cmr-T5,
5754 <cmr> load = cmr-default ]
5755 <bch> [ name = bch-T5,
5756 <bch> load = bch-default ]
5757 { encoding = T5,
5758 <cmr> family = cmr }
5759 <bch> family = bch }
5760 {

```

22 Contributed by Maciej Eder.


```

5761 <bch>    _ = {100,100},
5762 <bch>    \textbackslash = {150,200},
5763 <cmr>    \textbackslash = {200,300},
5764 <cmr>    \textquotedblleft = {200,600},
5765 <cmr>    \textquotedbl = {300,300},
5766 <bch>    \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5767 <cmr>    \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5768 <bch>    \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5769 <cmr>    \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5770 <bch>    \guillemotleft = {200,200}, \guillemotright = {150,300},
5771 <cmr>    \guillemotleft = {300,200}, \guillemotright = {100,400},
5772 <bch>    \textbraceleft = {200, }, \textbraceright = { ,300},
5773 <cmr>    \textbraceleft = {400,200}, \textbraceright = {200,400},
5774    \textless = {200,100}, \textgreater = {100,200}
5775 }
5776
5777 </cmr|bch>

```

Minion with lining numbers.

```

5778 <*pmn>
5779 \SetProtrusion
5780 [ name = pmnx-OT1,
5781 load = pmnj-default ]
5782 { encoding = OT1,
5783 family = pmnx }
5784 {
5785 1 = {230,180}
5786 }
5787
5788 \SetProtrusion
5789 [ name = pmnx-T1,
5790 load = pmnj-T1 ]
5791 { encoding = {T1,LY1},
5792 family = pmnx }
5793 {
5794 1 = {230,180}
5795 }
5796
5797 \SetProtrusion
5798 [ name = pmnx-T2A,
5799 load = pmnj-T2A ]
5800 { encoding = {T2A},
5801 family = pmnx }
5802 {
5803 1 = {230,180}
5804 }
5805
5806 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

5807 <*ptm>
5808 \SetProtrusion
5809 [ name = ptm-LY1,
5810 load = ptm-T1 ]
5811 { encoding = LY1,
5812 family = {ptm,ptmx,ptmj} }
5813 {
5814 _ = {100,100},
5815 \texttrademark = {100,100},
5816 \textregistered = {100,100},
5817 \textcopyright = {100,100},
5818 \textdegree = {300,300},
5819 \textminus = {200,200},
5820 \textellipsis = {150,200},

```

```

5821 % \texteuro = { , }, % ?
5822 \textcent = {100,100},
5823 \textquotesingle = {500,500},
5824 \textflorin = { 50, 70},
5825 \textdagger = {150,150},
5826 \textdaggerdbl = {100,100},
5827 \textperthousand = { , 50},
5828 \textbullet = {150,150},
5829 \textonesuperior = {100,100},
5830 \texttwosuperior = { 50, 50},
5831 \textthreesuperior = { 50, 50},
5832 \textperiodcentered = {300,300},
5833 \textplusminus = { 50, 80},
5834 \textmultiply = {100,100},
5835 \textdivide = { 50,150}

```

Remaining slots in the source file.

```

5836 }
5837
5838 </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

```

5839 \SetProtrusion
5840 <m-t> [ name = OT1-it ]
5841 <bch> [ name = bch-it ]
5842 <blg> [ name = blg-it,
5843 <blg> load = blg-default ]
5844 <cmr> [ name = cmr-it ]
5845 <pad> [ name = pad-it ]
5846 <pmn> [ name = pmn-it ]
5847 <ppl> [ name = ppl-it ]
5848 <ptm> [ name = ptm-it ]
5849 <ugm> [ name = ugm-it ]
5850 <m-t|bch|blg|pad|ugm> { encoding = OT1,
5851 <ppl|ptm> { encoding = {OT1,OT4},
5852 <bch> family = bch,
5853 <blg> family = blg,
5854 <pad> family = {pad,padx,padj},
5855 <ppl> family = {ppl,pplx,pplj},
5856 <ptm> family = {ptm,ptmx,ptmj},
5857 <ugm> family = ugm,
5858 <m-t|bch|pad|ppl|ptm> shape = {it,sl} }
5859 <blg|ugm> shape = it }
5860 <cmr|pmn> { }
5861 {
5862 <cmr> A = {100,100},
5863 <ptm> A = {100,50},
5864 <pad|pmn> A = {50, },
5865 <ugm> A = { ,150},
5866 <ppl> A = {50,50},
5867 <ptm> \AE = {100, },
5868 <pad|ppl> \AE = {50, },
5869 <cmr> B = {83,-40},

```

²³ Settings contributed by Hendrik Vogt.

5870 <pad|ppl|ptm> B = {50, },
5871 <pmn> B = {20,-50},
5872 <bch|ppl|ptm|ugm> C = {50, },
5873 <cmr> C = {165,-75},
5874 <pad> C = {100, },
5875 <pmn> C = {50,-50},
5876 <cmr> D = {75, -28},
5877 <pad|ppl|ptm> D = {50,50},
5878 <pmn> D = {20, },
5879 <cmr> E = {80,-55},
5880 <pad|ppl|ptm> E = {50, },
5881 <pmn> E = {20,-50},
5882 <cmr> F = {85,-80},
5883 <pad|ptm> F = {100, },
5884 <pmn> F = {10, },
5885 <ppl> F = {50, },
5886 <bch|ppl|ptm|ugm> G = {50, },
5887 <cmr> G = {153,-15},
5888 <pad> G = {100, },
5889 <pmn> G = {50,-50},
5890 <cmr> H = {73,-60},
5891 <pad|ppl|ptm> H = {50, },
5892 <cmr> I = {140,-120},
5893 <pad|ptm> I = {50, },
5894 <pmn> I = {20,-50},
5895 <cmr> J = {135,-80},
5896 <pad> J = {50, },
5897 <pmn> J = {20, },
5898 <ptm> J = {100, },
5899 <cmr> K = {70,-30},
5900 <pad|ppl|ptm> K = {50, },
5901 <pmn> K = {20, },
5902 <cmr> L = {87, 40},
5903 <pad|ppl|ptm> L = {50, },
5904 <pmn> L = {20,50},
5905 <ugm> L = { ,100},
5906 <cmr> M = {67,-45},
5907 <pmn> M = { , -30},
5908 <ptm> M = {50, },
5909 <cmr> N = {75,-55},
5910 <pmn> N = { , -30},
5911 <ptm> N = {50, },
5912 <bch|pmn|ppl|ptm> O = {50, },
5913 <cmr> O = {150,-30},
5914 <pad> O = {100, },
5915 <ugm> O = {70,50},
5916 <ppl|ptm> \OE = {50, },
5917 <pad> \OE = {100, },
5918 <cmr> P = {82,-50},
5919 <pad|ppl|ptm> P = {50, },
5920 <pmn> P = {20,-50},
5921 <bch|pmn|ppl|ptm> Q = {50, },
5922 <cmr> Q = {150,-30},
5923 <pad> Q = {100, },
5924 <ugm> Q = {70,50},
5925 <cmr> R = {75, 15},
5926 <pad|ppl|ptm> R = {50, },
5927 <pmn> R = {20, },
5928 <bch|pad|ppl|ptm> S = {50, },
5929 <cmr> S = {90,-65},
5930 <pmn> S = {20,-30},
5931 <bch|pad|ppl|ptm> \$ = {50, },
5932 <cmr> \$ = {100,-20},
5933 <pmn> \$ = {20,-30},
5934 <bch|pmn|ugm> T = {70, },

5935 <cmr> T = {220,-85},
5936 <pad|ppl|ptm> T = {100, },
5937 <cmr> U = {230,-55},
5938 <pad|ppl|ptm> U = {50, },
5939 <pmn> U = {50,-50},
5940 <cmr> V = {260,-60},
5941 <pad|pmn|ugm> V = {100, },
5942 <ppl|ptm> V = {100,50},
5943 <cmr> W = {185,-55},
5944 <pad|pmn|ugm> W = {100, },
5945 <ppl> W = {50, },
5946 <ptm> W = {100,50},
5947 <cmr> X = {70,-30},
5948 <ppl|ptm> X = {50, },
5949 <cmr> Y = {250,-60},
5950 <pmn> Y = {50, },
5951 <ppl> Y = {100,50},
5952 <ptm> Y = {100, },
5953 <cmr> Z = {90,-60},
5954 <pmn> Z = { , -50},
5955 <cmr> a = {150,-10},
5956 <cmr> b = {170, },
5957 <cmr> c = {173,-10},
5958 <cmr> d = {150,-55},
5959 <pmn> d = { , -50},
5960 <cmr> e = {180, },
5961 <cmr> f = { , -250},
5962 <pad|pmn> f = { , -100},
5963 <cmr> g = {150,-10},
5964 <cmr> h = {100, },
5965 <cmr> i = {210, },
5966 <pmn> i = { , -30},
5967 <cmr> j = { , -40},
5968 <pmn> j = { , -30},
5969 <cmr> k = {110,-50},
5970 <cmr> l = {240,-110},
5971 <pmn> l = { , -100},
5972 <cmr> m = {80, },
5973 <cmr> n = {115, },
5974 <bch> o = {50,50},
5975 <cmr> o = {155, },
5976 <bch> p = { , 50},
5977 <pmn> p = {-50, },
5978 <bch> q = {50, },
5979 <cmr> q = {170,-40},
5980 <cmr> r = {155,-40},
5981 <pmn> r = { , 50},
5982 <cmr> s = {130, },
5983 <bch> t = { , 50},
5984 <cmr> t = {230,-10},
5985 <cmr> u = {120, },
5986 <cmr> v = {140,-25},
5987 <pmn|ugm> v = {50, },
5988 <bch> w = { , 50},
5989 <cmr> w = {98,-20},
5990 <pmn|ugm> w = {50, },
5991 <cmr> x = {65,-40},
5992 <bch> y = { , 50},
5993 <cmr> y = {130,-20},
5994 <cmr> z = {110,-80},
5995 <cmr> 0 = {170,-85},
5996 <bch|ptm> 1 = {150,100},
5997 <cmr> 1 = {230,110},
5998 <pad> 1 = {150, },
5999 <pmn> 1 = {50, },

```

6000 <ppl>      1 = {100, },
6001 <ugm>      1 = {150,150},
6002 <cmr>      2 = {130,-70},
6003 <pad|ppl|ptm> 2 = {50, },
6004 <pmn>      2 = {-50, },
6005 <bch>      3 = {50, },
6006 <cmr>      3 = {140,-70},
6007 <pmn>      3 = {-100, },
6008 <ptm>      3 = {100,50},
6009 <bch>      4 = {100, },
6010 <cmr>      4 = {130,80},
6011 <pad>      4 = {150, },
6012 <ppl|ptm> 4 = {50, },
6013 <cmr>      5 = {160, },
6014 <ptm>      5 = {50, },
6015 <bch>      6 = {50, },
6016 <cmr>      6 = {175,-30},
6017 <bch|pad|ptm> 7 = {100, },
6018 <cmr>      7 = {250,-150},
6019 <pmn>      7 = {20, },
6020 <ppl>      7 = {50, },
6021 <cmr>      8 = {130,-40},
6022 <cmr>      9 = {155,-80},
6023 <m-t|cmr|pad|pmn|ppl> . = { ,500},
6024 <blg>      . = {400,600},
6025 <bch|ptm|ugm> . = { ,700},
6026 <blg>      {,}= {300,500},
6027 <m-t|pad|pmn|ppl> {,}= { ,500},
6028 <cmr>      {,}= { ,450},
6029 <bch|ugm>  {,}= { ,600},
6030 <ptm>      {,}= { ,700},
6031 <m-t|cmr|pad|ppl> : = { ,300},
6032 <bch|ugm>  : = { ,400},
6033 <pmn>      : = { ,200},
6034 <ptm>      : = { ,500},
6035 <m-t|cmr|pad|ppl> ; = { ,300},
6036 <bch|ugm>  ; = { ,400},
6037 <pmn>      ; = { ,200},
6038 <ptm>      ; = { ,500},
6039 <ptm>      ! = { ,100},
6040 <bch>      ? = { ,200},
6041 <ptm>      ? = { ,100},
6042 <ppl>      ? = { ,300},
6043 <pmn>      " = {400,200},
6044 <m-t|pad|pmn|ppl|ptm> & = {50,50},
6045 <bch>      & = { ,80},
6046 <cmr>      & = {130,30},
6047 <ugm>      & = {50,100},
6048 <m-t|pad|pmn> \% = {100, },
6049 <cmr>      \% = {180,50},
6050 <bch>      \% = {50,50},
6051 <ppl|ptm>  \% = {100,100},
6052 <ugm>      \% = {100,50},
6053 <m-t|pmn|ppl> * = {200,200},
6054 <bch>      * = {300,200},
6055 <cmr>      * = {380,20},
6056 <pad>      * = {500,100},
6057 <ptm|ugm> * = {400,200},
6058 <m-t|pmn|ppl> + = {150,200},
6059 <cmr>      + = {180,200},
6060 <bch|ugm>  + = {250,250},
6061 <pad|ptm>  + = {250,200},
6062 <m-t|pad|pmn|ppl> @ = {50,50},
6063 <bch>      @ = {80,50},
6064 <cmr>      @ = {180,10},

```

```

6065 <ptm> @ = {150,150},
6066 <m-t|bch|ugm> ~ = {150,150},
6067 <cmr|pad|pmn|ppl|ptm> ~ = {200,150},
6068 <ugm> {=} = {200,200},
6069 <m-t|bch|pad|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
6070 <cmr> ( = {300, }, ) = { ,70},
6071 <m-t|pad|ppl|ptm|ugm> / = {100,200},
6072 <cmr> / = {100,100},
6073 <bch> / = { ,150},
6074 <pmn> / = {100,150},
6075 <m-t> - = {300,300},
6076 <bch|pad> - = {300,400},
6077 <pmn> - = {200,300},
6078 <cmr> - = {500,300},
6079 <ppl> - = {300,500},
6080 <ptm> - = {500,500},
6081 <ugm> - = {400,700},
6082 <blg> - = {0,300},
6083 <m-t|pmn> \textendash = {200,200}, \textemdash = {150,150},
6084 <bch> \textendash = {200,300}, \textemdash = {150,200},
6085 <cmr> \textendash = {500,300}, \textemdash = {400,170},
6086 <pad|ppl|ptm|ugm> \textendash = {300,300}, \textemdash = {200,200},
6087 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
6088 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
6089 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
6090 <pad> \textquoteleft = {800,200}, \textquoteright = {800,200},
6091 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
6092 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
6093 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
6094 <blg> \textquotedblright = {300,300}
6095 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
6096 <pad> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
6097 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
6098 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
6099 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
6100 }
6101
6102 (*cmr|pmn)
6103 \SetProtrusion
6104 <cmr> [ name = cmr-it-OT1,
6105 <pmn> [ name = pmnj-it-OT1,
6106 <cmr> load = cmr-it ]
6107 <pmn> load = pmnj-it ]
6108 <cmr> { encoding = {OT1,OT4},
6109 <pmn> { encoding = OT1,
6110 <cmr> family = cmr,
6111 <pmn> family = pmnj,
6112 <cmr> shape = it }
6113 <pmn> shape = {it,s1} }
6114 {
6115 <cmr> \AE = {100, },
6116 <pmn> \AE = { , -50},
6117 <cmr> \OE = {100, },
6118 <pmn> \OE = {50, }
6119 (*cmr)
6120 "00 = {200,150}, % \Gamma
6121 "01 = {150,100}, % \Delta
6122 "02 = {150, 50}, % \Theta
6123 "03 = {150, 50}, % \Lambda
6124 "04 = {100,100}, % \Xi
6125 "05 = {100,100}, % \Pi
6126 "06 = {100, 50}, % \Sigma
6127 "07 = {200,150}, % \Upsilon
6128 "08 = {150, 50}, % \Phi
6129 "09 = {150,100}, % \Psi

```

```

6130 "OA = { 50, 50} % \Omega
6131 </cmr>
6132 }
6133
6134 </cmr|pmn>
6135 \SetProtrusion
6136 <m-t> [ name = T1-it-default,
6137 <bch> [ name = bch-it-T1,
6138 <blg> [ name = blg-it-T1,
6139 <cmr> [ name = cmr-it-T1,
6140 <pad> [ name = pad-it-T1,
6141 <pmn> [ name = pmnj-it-T1,
6142 <ppl> [ name = ppl-it-T1,
6143 <ptm> [ name = ptm-it-T1,
6144 <ugm> [ name = ugm-it-T1,
6145 <m-t> load = OT1-it ]
6146 <bch> load = bch-it ]
6147 <blg> load = blg-T1 ]
6148 <cmr> load = cmr-it ]
6149 <pmn> load = pmnj-it ]
6150 <pad> load = pad-it ]
6151 <ppl> load = ppl-it ]
6152 <ptm> load = ptm-it ]
6153 <ugm> load = ugm-it ]
6154 <m-t|bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
6155 <blg|ptm|ugm> { encoding = T1,
6156 <bch> family = bch,
6157 <blg> family = blg,
6158 <cmr> family = cmr,
6159 <pmn> family = pmnj,
6160 <pad> family = {pad,padx,padj},
6161 <ppl> family = {ppl,pplx,pplj},
6162 <ptm> family = {ptm,ptmx,ptmj},
6163 <ugm> family = ugm,
6164 <m-t|bch|pad|pmn|ppl|ptm> shape = {it,sl} }
6165 <blg|cmr|ugm> shape = it }
6166 {
6167 <m-t|bch|pmn> _ = { ,100},
6168 <blg> _ = {0,300},
6169 <cmr|ugm> _ = {100,200},
6170 <pad|ppl|ptm> _ = {100,100},
6171 <blg> . = {400,600},
6172 <blg> {,}= {300,500},
6173 <cmr> \AE = {100, },
6174 <pmn> \AE = { , -50},
6175 <bch|pmn> \OE = { 50, },
6176 <cmr> \OE = {100, },
6177 <pmn> O31 = { , -100}, % ffl
6178 <cmr|ptm> 156 = {100, }, % IJ
6179 <pad> 156 = {50, }, % IJ
6180 <pmn> 156 = {20, }, % IJ
6181 <pmn> 188 = { , -30}, % ij
6182 <pmn> \v t = { ,100},
6183 <m-t|pad|ppl|ptm> \textbackslash = {100,200},
6184 <cmr|ugm> \textbackslash = {300,300},
6185 <bch> \textbackslash = {150,150},
6186 <pmn> \textbackslash = {100,150},
6187 <ugm> \textbar = {200,200},
6188 <cmr> \textquotedblleft = {500,300},
6189 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
6190 <blg> \textquotedbl = {300,300}, \textquotedblleft = {300,300},
6191 <blg> \textquotedblright = {300,300}, \quotedblbase = {200,600},
6192 <m-t|ptm> \quotesinglbase = {300,700}, \quotedblbase = {400,500},
6193 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
6194 <bch|pmn> \quotesinglbase = {200,500}, \quotedblbase = {150,500},

```

```

6195 <pad|ppl> \quotesinglbase = {500,500}, \quotedblbase = {400,400},
6196 <ugm> \quotesinglbase = {300,700}, \quotedblbase = {300,500},
6197 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsingright = {300,500},
6198 <bch|pmn> \guilsinglleft = {300,400}, \guilsingright = {200,500},
6199 <cmr> \guilsinglleft = {500,300}, \guilsingright = {400,400},
6200 <pad> \guilsinglleft = {500,400}, \guilsingright = {300,500},
6201 <ugm> \guilsinglleft = {400,400}, \guilsingright = {300,600},
6202 <m-t|ppl> \guillemotleft = {300,300}, \guillemotright = {300,300},
6203 <bch|pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
6204 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
6205 <pad> \guillemotleft = {300,300}, \guillemotright = {200,400},
6206 <ptm> \guillemotleft = {300,400}, \guillemotright = {200,400},
6207 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
6208 <m-t|pad|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
6209 <cmr|ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
6210 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
6211 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
6212 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
6213 <cmr|pad|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
6214 <bch|pmn> \textless = {100, }, \textgreater = { ,100},
6215 <cmr|pad|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
6216 <pmn> \textvisiblespace = {100,100}
6217 }
6218
6219 <*m-t|cmr|pmn>
6220 \SetProtrusion
6221 <m-t> [ name = T2A-it-default,
6222 <cmr> [ name = cmr-it-T2A,
6223 <pmn> [ name = pmnj-it-T2A,
6224 <m-t> load = OT1-it ]
6225 <cmr> load = cmr-it ]
6226 <pmn> load = pmnj-it ]
6227 { encoding = T2A,
6228 <cmr> family = cmr,
6229 <pmn> family = pmnj,
6230 <m-t|pmn> shape = {it,sl} }
6231 <cmr> shape = it }
6232 {
6233 <cmr> \CYRA = {100,50},
6234 <pmn> \CYRA = {50, },
6235 <cmr> \CYRB = {50, },
6236 <cmr> \CYRV = {50, },
6237 <pmn> \CYRV = {20,-50},
6238 <cmr> \CYRG = {100, },
6239 <pmn> \CYRG = {10, },
6240 <cmr> \CYRD = {50, },
6241 <cmr> \CYRE = {50, },
6242 <pmn> \CYRE = {20,-50},
6243 <cmr> \CYRZH = {50, },
6244 <cmr> \CYRZ = {50, },
6245 <pmn> \CYRZ = {20,-50},
6246 <cmr> \CYRI = {50, },
6247 <pmn> \CYRI = { , -30},
6248 <cmr> \CYRISHRT = {50, },
6249 <cmr> \CYRK = {50, },
6250 <pmn> \CYRK = {20, },
6251 <cmr> \CYRL = {50, },
6252 <cmr> \CYRM = {50, },
6253 <pmn> \CYRM = { , -30},
6254 <cmr> \CYRN = {50, },
6255 <cmr> \CYRO = {100, },
6256 <pmn> \CYRO = {50, },
6257 <cmr> \CYRP = {50, },
6258 <cmr> \CYRR = {50, },
6259 <pmn> \CYRR = {20,-50},

```



```

6260 <cmr> \CYRS = {100, },
6261 <pmn> \CYRS = {50, },
6262 <cmr> \CYRT = {100, },
6263 <pmn> \CYRT = {70, },
6264 <cmr> \CYRU = {100, },
6265 <pmn> \CYRU = {50, },
6266 <cmr> \CYRF = {100, },
6267 <cmr> \CYRH = {50, },
6268 <cmr> \CYRC = {50, },
6269 <cmr> \CYRCH = {100, },
6270 <cmr> \CYRSH = {50, },
6271 <cmr> \CYRSHCH = {50, },
6272 <cmr> \CYRHRDSN = {100, },
6273 <cmr> \CYRERY = {50, },
6274 <cmr> \CYRSFTSN = {50, },
6275 <cmr> \CYREREV = {50, },
6276 <cmr> \CYRYU = {50, },
6277 <cmr> \CYRYA = {50, },
6278 <pmn> \CYRYA = { ,20},
6279 <pmn> \cyrr = {-50, },
6280 <m-t|pmn> _ = { ,100},
6281 <cmr> _ = {100,200},
6282 <pmn> 031 = { , -100}, % ff1
6283 <pmn> \v t = { ,100},
6284 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
6285 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
6286 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
6287 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
6288 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
6289 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
6290 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
6291 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
6292 <pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
6293 <cmr> \textquotedblleft = {500,300},
6294 <cmr> \textless = {300,100}, \textgreater = {200,100}
6295 <pmn> \textless = {100, }, \textgreater = { ,100}
6296 }
6297
6298 </m-t|cmr|pmn>
6299 <*-t|ptm>
6300 \SetProtrusion
6301 <m-t> [ name = QX-it-default,
6302 <ptm> [ name = ptm-it-QX,
6303 <m-t> load = OT1-it ]
6304 <ptm> load = ptm-it ]
6305 { encoding = {QX},
6306 <ptm> family = {ptm,ptmx,ptmj},
6307 shape = {it,s1} }
6308 {
6309 <ptm> 009 = { , 50}, % fk
6310 {=} = {100,100},
6311 <m-t> \textunderscore = {100,100},
6312 <ptm> \textunderscore = {100,150},
6313 \textbackslash = {100,200},
6314 \quotedblbase = {300,400},
6315 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
6316 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
6317 \textexclamdown = {200, }, \textquestiondown = {200, },
6318 \textbraceleft = {200,100}, \textbraceright = {200,200},
6319 \textless = {100,100}, \textgreater = {100,100},
6320 \textminus = {200,200}, \textdegree = {300,150},
6321 <m-t> \copyright = {100,100}, \textregistered = {100,100}
6322 <ptm> \textregistered = {100,150}, \copyright = {100,150},
6323 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
6324 <ptm> \textpi = { 50, 80}, \textmu = { , 80},

```

```

6325 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
6326 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
6327 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
6328 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
6329 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
6330 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
6331 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
6332 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
6333 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
6334 <ptm> \textperthousand = { ,50}
6335 }
6336
6337 </m-t|ptm>
6338 <*cmr|bch>
6339 \SetProtrusion
6340 <cmr> [ name = cmr-it-T5,
6341 <cmr> load = cmr-it ]
6342 <bch> [ name = bch-it-T5,
6343 <bch> load = bch-it ]
6344 { encoding = T5,
6345 <bch> family = bch,
6346 <cmr> family = cmr,
6347 shape = it }
6348 {
6349 <bch> _ = { ,100},
6350 <cmr> _ = {100,200},
6351 <bch> \textbackslash = {150,150},
6352 <cmr> \textbackslash = {300,300},
6353 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
6354 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
6355 <bch> \guilsinglleft = {300,400}, \guilsingright = {200,500},
6356 <cmr> \guilsinglleft = {500,300}, \guilsingright = {400,400},
6357 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
6358 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
6359 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
6360 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
6361 <bch> \textless = {100, }, \textgreater = { ,100},
6362 <cmr> \textless = {300,100}, \textgreater = {200,100}
6363 }
6364
6365 </cmr|bch>

```

Slanted is very similar to italic.

```

6366 <*cmr>
6367 \SetProtrusion
6368 [ name = cmr-sl,
6369 load = cmr-it-OT1 ]
6370 { encoding = {OT1,OT4},
6371 family = cmr,
6372 shape = sl }
6373 {
6374 L = { ,50},
6375 f = { , -50},
6376 - = {300, },
6377 \textendash = {400, }, \textemdash = {300, }
6378 }
6379
6380 \SetProtrusion
6381 [ name = cmr-sl-T1,
6382 load = cmr-it-T1 ]
6383 { encoding = {T1,LY1},
6384 family = cmr,
6385 shape = sl }
6386 {
6387 L = { ,50},

```

```

6388     f = { , -50},
6389     - = {300, },
6390     \textendash = {400, }, \textemdash = {300, }
6391   }
6392
6393 \SetProtrusion
6394   [ name = cmr-sl-T2A,
6395     load = cmr-it-T2A ]
6396   { encoding = T2A,
6397     family = cmr,
6398     shape = sl }
6399   {
6400     L = { ,50},
6401     f = { , -50},
6402     - = {300, },
6403     \textendash = {400, }, \textemdash = {300, }
6404   }
6405
6406 \SetProtrusion
6407   [ name = cmr-sl-T5,
6408     load = cmr-it-T5 ]
6409   { encoding = T5,
6410     family = cmr,
6411     shape = sl }
6412   {
6413     L = { ,50},
6414     f = { , -50},
6415     - = {300, },
6416     \textendash = {400, }, \textemdash = {300, }
6417   }
6418
6419 \SetProtrusion
6420   [ name = lmr-it-T1,
6421     load = cmr-it-T1 ]
6422   { encoding = {T1,LY1},
6423     family = lmr,
6424     shape = {it,sl} }
6425   {
6426     \textquotedblleft = { ,200}, \textquotedblright = { ,200},
6427     \quotesinglbase = { ,400}, \quotedblbase = { ,500}
6428   }
6429

```

Oldstyle numerals are slightly different.

```

6430 \SetProtrusion
6431   [ name = cmr(oldstyle)-it,
6432     load = cmr-it-T1 ]
6433   { encoding = T1,
6434     family = {hfor,cmor},
6435     shape = {it,sl} }
6436   {
6437     1 = {250, 50},
6438     2 = {150, -100},
6439     3 = {100, -50},
6440     4 = {150, 150},
6441     6 = {200, },
6442     7 = {200, 50},
6443     8 = {150, -50},
6444     9 = {100, 50}
6445   }
6446
6447 </cmr>
6448 < *pmn >
6449 \SetProtrusion
6450   [ name = pmnx-it,

```

```

6451     load    = pmnj-it ]
6452     { encoding = OT1,
6453       family   = pmnx,
6454       shape    = {it,s1} }
6455     {
6456       1 = {100,150}
6457     }
6458
6459 \SetProtrusion
6460     [ name    = pmnx-it-T1,
6461       load    = pmnj-it-T1 ]
6462     { encoding = {T1,LY1},
6463       family   = pmnx,
6464       shape    = {it,s1} }
6465     {
6466       1 = {100,150}
6467     }
6468
6469 \SetProtrusion
6470     [ name    = pmnx-it-T2A,
6471       load    = pmnj-it-T2A ]
6472     { encoding = {T2A},
6473       family   = pmnx,
6474       shape    = {it,s1} }
6475     {
6476       1 = {100,150}
6477     }
6478
6479 </pmn>
6480 <*ptm>
6481 \SetProtrusion
6482     [ name    = ptm-it-LY1,
6483       load    = ptm-it-T1 ]
6484     { encoding = {LY1},
6485       family   = {ptm,ptmx,ptmj},
6486       shape    = {it,s1} }
6487     {
6488       - = {100,100},
6489       \texttrademark = {100,100},
6490       \textregistered = {100,100},
6491       \textcopyright = {100,100},
6492       \textdegree = {300,100},
6493       \textminus = {200,200},
6494       \textellipsis = {100,200},
6495 % \texteuro = { , }, % ?
6496       \textcent = {100,100},
6497       \textquotesingle = {500, },
6498       \textflorin = {100, 70},
6499       \textdagger = {150,150},
6500       \textdaggerdbl = {100,100},
6501       \textbullet = {150,150},
6502       \textonesuperior = {150,100},
6503       \texttwosuperior = {150, 50},
6504       \textthreesuperior = {150, 50},
6505       \textparagraph = {100, },
6506       \textperiodcentered = {500,300},
6507       \textonequarter = { 50, },
6508       \textonehalf = { 50, },
6509       \textplusminus = {100,100},
6510       \textmultiply = {150,150},
6511       \textdivide = {150,150}
6512     }
6513
6514 </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).

```

6515 <*(blg|ugm)>
6516 \SetProtrusion
6517 <m-t> [ name = OT1-sc,
6518 <bch> [ name = bch-sc,
6519 <cmr> [ name = cmr-sc-OT1,
6520 <pad> [ name = pad-sc,
6521 <pmn> [ name = pmnj-sc,
6522 <ppl> [ name = ppl-sc,
6523 <ptm> [ name = ptm-sc,
6524 <m-t> load = default ]
6525 <bch> load = bch-default ]
6526 <cmr> load = cmr-OT1 ]
6527 <pad> load = pad-default ]
6528 <pmn> load = pmnj-default ]
6529 <ppl> load = ppl-default ]
6530 <ptm> load = ptm-default ]
6531 <m-t|bch|pad|pmn> { encoding = OT1,
6532 <cmr|ppl|ptm> { encoding = {OT1,OT4},
6533 <bch> family = bch,
6534 <cmr> family = cmr,
6535 <pad> family = {pad,padx,padj},
6536 <pmn> family = pmnj,
6537 <ppl> family = {ppl,pplx,pplj},
6538 <ptm> family = {ptm,ptmx,ptmj},
6539 shape = sc }
6540 {
6541 a = {50,50},
6542 <cmr|pad|ppl|ptm> \ae = {50, },
6543 <bch|pmn> c = {50, },
6544 <bch|pad|pmn> d = { ,50},
6545 <m-t|bch|cmr|pad|pmn|ptm> f = { ,50},
6546 <bch|pad|pmn> g = {50, },
6547 <m-t|cmr|pad|pmn|ppl|ptm> j = {50, },
6548 <bch> j = {100, },
6549 <m-t|bch|cmr|pad|pmn|ppl> l = { ,50},
6550 <ptm> l = { ,80},
6551 <m-t|bch|cmr|pad|pmn|ppl> 013 = { ,50}, % fl
6552 <ptm> 013 = { ,80}, % fl
6553 <bch|pad|pmn> o = {50,50},
6554 <pad|pmn> \oe = {50, },
6555 <ppl> p = { 0, 0},
6556 <bch|pad|pmn> q = {50,70},
6557 <ppl> q = { 0, },
6558 <m-t|cmr|pad|pmn|ppl|ptm> r = { , 0},
6559 t = {50,50},
6560 <m-t|bch|cmr|pad|pmn|ppl> y = {50,50}
6561 <ptm> y = {80,80}
6562 }
6563
6564 \SetProtrusion
6565 <m-t> [ name = T1-sc,
6566 <bch> [ name = bch-sc-T1,
6567 <cmr> [ name = cmr-sc-T1,
6568 <pad> [ name = pad-sc-T1,
6569 <pmn> [ name = pmnj-sc-T1,
6570 <ppl> [ name = ppl-sc-T1,
6571 <ptm> [ name = ptm-sc-T1,
6572 <m-t> load = T1-default ]
6573 <bch> load = bch-T1 ]

```

```

6574 <cmr>    load    = cmr-T1    ]
6575 <pad>    load    = pad-T1    ]
6576 <pmn>    load    = pmnj-T1   ]
6577 <ppl>    load    = ppl-T1    ]
6578 <ptm>    load    = ptm-T1    ]
6579 { encoding = {T1,LY1},
6580 <bch>    family  = bch,
6581 <cmr>    family  = cmr,
6582 <pad>    family  = {pad,padx,padj},
6583 <pmn>    family  = pmnj,
6584 <ppl>    family  = {ppl,pplx,pplj},
6585 <ptm>    family  = {ptm,ptmx,ptmj},
6586 shape    = sc }
6587 {
6588 a = {50,50},
6589 <cmr|pad|ppl|ptm> \ae = {50, },
6590 <bch|pmn> c = {50, },
6591 <bch|pad|pmn> d = { ,50},
6592 <m-t|bch|cmr|pad|pmn|ptm> f = { ,50},
6593 <bch|pad|pmn> g = {50, },
6594 <m-t|cmr|pad|pmn|ppl|ptm> j = {50, },
6595 <bch> j = {100, },
6596 <m-t|bch|cmr|pad|pmn|ppl> l = { ,50},
6597 <ptm> l = { ,80},
6598 <m-t|bch|cmr|pad|pmn|ppl> 029 = { ,50}, % f1
6599 <ptm> 029 = { ,80}, % f1
6600 <bch|pad|pmn> o = {50,50},
6601 <bch|pad|pmn> \oe = {50, },
6602 <ppl> p = { 0, 0},
6603 <bch|pad|pmn> q = {50,70},
6604 <ppl> q = { 0, },
6605 <m-t|cmr|pad|pmn|ppl|ptm> r = { , 0},
6606 t = {50,50},
6607 <m-t|bch|cmr|pad|pmn|ppl> y = {50,50}
6608 <ptm> y = {80,80}
6609 }
6610
6611 </!(b|g|ugm)>
6612 <*-t|cmr>
6613 \SetProtrusion
6614 <m-t> [ name    = T2A-sc,
6615 <cmr> [ name    = cmr-sc-T2A,
6616 <m-t> load    = T2A-default ]
6617 <cmr> load    = cmr-T2A    ]
6618 { encoding = T2A,
6619 <cmr> family  = cmr,
6620 shape    = sc }
6621 {
6622 \cyra = {50,50},
6623 \cyrg = { ,50},
6624 \cyrt = {50,50},
6625 \cyry = { ,50}
6626 }
6627
6628 </m-t|cmr>
6629 <*-t>
6630 \SetProtrusion
6631 [ name    = QX-sc,
6632 load    = QX-default ]
6633 { encoding = QX,
6634 shape    = sc }
6635 {
6636 a = {50,50},
6637 f = { ,50},
6638 j = {50, },

```

```

6639     l = { ,50},
6640     013 = { ,50}, % fl
6641     r = { , 0},
6642     t = {50,50},
6643     y = {50,50}
6644   }
6645
6646 </m-t>
6647 <*cmr|bch>
6648 \SetProtrusion
6649 <bch> [ name      = bch-sc-T5,
6650 <bch>   load      = bch-T5 ]
6651 <cmr> [ name      = cmr-sc-T5,
6652 <cmr>   load      = cmr-T5 ]
6653 { encoding = T5,
6654 <bch>   family   = bch,
6655 <cmr>   family   = cmr,
6656   shape   = sc }
6657 {
6658   a = {50,50},
6659 <bch>   c = {50, },
6660 <bch>   d = { ,50},
6661   f = { ,50},
6662 <bch>   g = {50, },
6663 <bch>   j = {100, },
6664 <cmr>   j = {50, },
6665   l = { ,50},
6666 <bch>   o = {50,50},
6667 <bch>   q = { 0, },
6668 <cmr>   r = { , 0},
6669   t = {50,50},
6670   y = {50,50}
6671 }
6672
6673 </cmr|bch>
6674 <*pmn>
6675 \SetProtrusion
6676 [ name      = pmnx-sc,
6677   load      = pmnj-sc ]
6678 { encoding = OT1,
6679   family   = pmnx,
6680   shape    = sc }
6681 {
6682   1 = {230,180}
6683 }
6684
6685 \SetProtrusion
6686 [ name      = pmnx-sc-T1,
6687   load      = pmnj-sc-T1 ]
6688 { encoding = {T1,LY1},
6689   family   = pmnx,
6690   shape    = sc }
6691 {
6692   1 = {230,180}
6693 }
6694

```

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`.

```

6695 \SetProtrusion
6696 [ name      = pmnj-scit,
6697   load      = pmnj-it ]

```

```

6698 { encoding = OT1,
6699     family   = pmnj,
6700     shape    = {scit,si} }
6701 {
6702     a = {50, },
6703     \ae = { , -50},
6704     b = {20, -50},
6705     c = {50, -50},
6706     d = {20, 0},
6707     e = {20, -50},
6708     f = {10, 0},
6709     012 = {10, -50}, % fi
6710     013 = {10, -50}, % fl
6711     014 = {10, -50}, % ffi
6712     015 = {10, -50}, % ffl
6713     g = {50, -50},
6714     i = {20, -50},
6715     j = {20, 0},
6716     k = {20, },
6717     l = {20, 50},
6718     m = { , -30},
6719     n = { , -30},
6720     o = {50, },
6721     \oe = {50, -50},
6722     p = {20, -50},
6723     q = {50, },
6724     r = {20, 0},
6725     s = {20, -30},
6726     t = {70, },
6727     u = {50, -50},
6728     v = {100, },
6729     w = {100, },
6730     y = {50, },
6731     z = { , -50}
6732 }
6733
6734 \SetProtrusion
6735 [ name   = pmnj-scit-T1,
6736   load   = pmnj-it-T1 ]
6737 { encoding = {T1,LY1},
6738   family   = pmnj,
6739   shape    = {scit,si} }
6740 {
6741     a = {50, },
6742     \ae = { , -50},
6743     b = {20, -50},
6744     c = {50, -50},
6745     d = {20, 0},
6746     e = {20, -50},
6747     f = {10, 0},
6748     028 = {10, -50}, % fi
6749     029 = {10, -50}, % fl
6750     030 = {10, -50}, % ffi
6751     031 = {10, -50}, % ffl
6752     g = {50, -50},
6753     i = {20, -50},
6754     188 = {20, 0}, % ij
6755     j = {20, 0},
6756     k = {20, },
6757     l = {20, 50},
6758     m = { , -30},
6759     n = { , -30},
6760     o = {50, },
6761     \oe = {50, -50},
6762     p = {20, -50},

```



```

6763   q = {50, },
6764   r = {20, 0},
6765   s = {20,-30},
6766   t = {70, },
6767   u = {50,-50},
6768   v = {100, },
6769   w = {100, },
6770   y = {50, },
6771   z = { , -50}
6772   }
6773
6774 \SetProtrusion
6775   [ name      = pmnx-scit,
6776     load      = pmnj-scit ]
6777   { encoding = OT1,
6778     family   = pmnx,
6779     shape    = {scit,si} }
6780   {
6781     1 = {100,150}
6782   }
6783
6784 \SetProtrusion
6785   [ name      = pmnx-scit-T1,
6786     load      = pmnj-scit-T1 ]
6787   { encoding = {T1,LY1},
6788     family   = pmnx,
6789     shape    = {scit,si} }
6790   {
6791     1 = {100,150}
6792   }
6793
6794 </pmn>

```

15.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino. Anybody?

```

6795 \SetProtrusion
6796 <m-t> [ name      = textcomp ]
6797 <bch> [ name      = bch-textcomp ]
6798 <blg> [ name      = blg-textcomp ]
6799 <cmr> [ name      = cmr-textcomp ]
6800 <pad> [ name      = pad-textcomp ]
6801 <pmn> [ name      = pmn-textcomp ]
6802 <ppl> [ name      = ppl-textcomp ]
6803 <ptm> [ name      = ptm-textcomp ]
6804 <ugm> [ name      = ugm-textcomp ]
6805 <m-t> { encoding = TS1 }
6806 <!m-t> { encoding = TS1,
6807   <bch>   family = bch }
6808   <blg>   family = blg }
6809   <cmr>   family = cmr }
6810   <pad>   family = {pad,padx,padj} }
6811   <pmn>   family = {pmnx,pmnj} }
6812   <ppl>   family = {ppl,pplx,pplj} }
6813   <ptm>   family = {ptm,ptmx,ptmj} }
6814   <ugm>   family = ugm }
6815   {
6816   <blg>   \textquotestraightbase = {400,500},
6817   <cmr>   \textquotestraightbase = {300,300},
6818   <pad|pmn> \textquotestraightbase = {400,400},
6819   <blg>   \textquotestraightdblbase = {300,400},
6820   <cmr|pmn> \textquotestraightdblbase = {300,300},
6821   <pad>   \textquotestraightdblbase = {400,400},

```

```

6822 <bch|cmr|pad|pmn|ugm> \texttwelveudash = {200,200},
6823 <bch|cmr|pad|pmn> \textthreequartersemdash = {150,150},
6824 <ugm> \textthreequartersemdash = {200,200},
6825 <blg> \textquotesingle = {500,600},
6826 <cmr|pmn> \textquotesingle = {300,400},
6827 <pad> \textquotesingle = {400,500},
6828 <ptm> \textquotesingle = {500,500},
6829 <ugm> \textquotesingle = {300,500},
6830 <bch|cmr|pmn> \textasteriskcentered = {200,300},
6831 <blg> \textasteriskcentered = {150,200},
6832 <pad> \textasteriskcentered = {300,300},
6833 <ugm> \textasteriskcentered = {100,200},
6834 <pmn> \textfractionsolidus = {-200,-200},
6835 <cmr> \textoneoldstyle = {100,100},
6836 <pmn> \textoneoldstyle = { , 50},
6837 <cmr> \textthreeoldstyle = { , 50},
6838 <pad|pmn> \textthreeoldstyle = { 50, },
6839 <cmr> \textfouroldstyle = { 50, 50},
6840 <pad|pmn> \textfouroldstyle = { 50, },
6841 <cmr|pad|pmn> \textsevenoldstyle = { 50, 80},
6842 <cmr> \textlangle = {400, },
6843 <cmr> \textrangle = { ,400},
6844 <m-t|bch|pmn|ptm> \textminus = {200,200},
6845 <cmr|pad|ppl> \textminus = {300,300},
6846 <blg|ugm> \textminus = {250,300},
6847 <bch|pad|pmn> \textlbrackdbl = {100, },
6848 <blg> \textlbrackdbl = {200, },
6849 <bch|pad|pmn> \textrbrackdbl = { ,100},
6850 <blg> \textrbrackdbl = { ,200},
6851 <pmn> \textasciigrave = {200,500},
6852 <bch|blg|cmr|pad|pmn> \texttildelow = {200,250},
6853 <pmn> \textasciibreve = {300,400},
6854 <pmn> \textasciicaron = {300,400},
6855 <pmn> \textacutedbl = {200,300},
6856 <pmn> \textgravedbl = {150,300},
6857 <bch|pmn|ugm> \textdagger = { 80, 80},
6858 <blg> \textdagger = {200,200},
6859 <cmr|pad> \textdagger = {100,100},
6860 <ptm> \textdagger = {150,150},
6861 <blg> \textdaggerdbl = {150,150},
6862 <cmr|pad|pmn> \textdaggerdbl = { 80, 80},
6863 <ptm> \textdaggerdbl = {100,100},
6864 <bch> \textbardbl = {100,100},
6865 <blg|ugm> \textbardbl = {150,150},
6866 <bch> \textbullet = {200,200},
6867 <blg> \textbullet = {400,500},
6868 <cmr|pad|pmn> \textbullet = { ,100},
6869 <ptm> \textbullet = {150,150},
6870 <ugm> \textbullet = { 50,100},
6871 <bch|cmr|pmn> \textcelsius = { 50, },
6872 <pad> \textcelsius = { 80, },
6873 <bch> \textflorin = { 50, 50},
6874 <blg> \textflorin = {100,100},
6875 <pad|ugm> \textflorin = { ,100},
6876 <pmn> \textflorin = { 50,100},
6877 <ptm> \textflorin = { 50, 70},
6878 <cmr> \textcolonmonetary = { , 50},
6879 <pad|pmn> \textcolonmonetary = { 50, },
6880 <pmn> \textinterrobang = { ,100},
6881 <pmn> \textinterrobangdown = {100, },
6882 <m-t|pad|ptm> \texttrademark = {100,100},
6883 <bch> \texttrademark = {150,150},
6884 <blg|cmr|ppl> \texttrademark = {200,200},
6885 <pmn> \texttrademark = { 50, 50},
6886 <ugm> \texttrademark = {100,150},

```

```

6887 <bch|ugm> \textcent = { 50, },
6888 <ptm> \textcent = {100,100},
6889 <bch> \textsterling = { 50, },
6890 <ugm> \textsterling = { , 50},
6891 <bch> \textbrokenbar = {200,200},
6892 <blg> \textbrokenbar = {250,250},
6893 <ugm> \textbrokenbar = {200,300},
6894 <pmn> \textasciidieresis = {300,400},
6895 <m-t|bch|cmr|pad|ptm|ugm> \textcopyright = {100,100},
6896 <pmn> \textcopyright = {100,150},
6897 <ppl> \textcopyright = {200,200},
6898 <bch|cmr|ugm> \textordfeminine = {100,200},
6899 <pad|pmn> \textordfeminine = {200,200},
6900 <bch|cmr|pad|pmn|ugm> \textlnot = {200, },
6901 <blg> \textlnot = {200,100},
6902 <m-t|bch|cmr|pad|ptm|ugm> \textregistered = {100,100},
6903 <pmn> \textregistered = { 50,150},
6904 <ppl> \textregistered = {200,200},
6905 <pmn> \textasciimacron = {150,200},
6906 <m-t|ppl|ptm> \textdegree = {300,300},
6907 <bch> \textdegree = {150,200},
6908 <blg|ugm> \textdegree = {200,200},
6909 <cmr|pad> \textdegree = {400,400},
6910 <pmn> \textdegree = {150,400},
6911 <bch|cmr|pad|pmn|ugm> \textpm = {150,200},
6912 <blg> \textpm = {100,100},
6913 <ptm> \textpm = { 50, 80},
6914 <bch|blg|ugm> \texttwosuperior = {100,200},
6915 <cmr> \texttwosuperior = { 50,100},
6916 <pad|pmn> \texttwosuperior = {200,200},
6917 <ptm> \texttwosuperior = { 50, 50},
6918 <bch|blg|ugm> \textthreesuperior = {100,200},
6919 <cmr> \textthreesuperior = { 50,100},
6920 <pad|pmn> \textthreesuperior = {200,200},
6921 <ptm> \textthreesuperior = { 50, 50},
6922 <pmn> \textasciacute = {300,400},
6923 <bch|ugm> \textmu = { ,100},
6924 <bch|pad|pmn> \textparagraph = { ,100},
6925 <bch|cmr|pad|pmn> \textperiodcentered = {300,400},
6926 <blg> \textperiodcentered = {400,500},
6927 <ptm> \textperiodcentered = {300,300},
6928 <ugm> \textperiodcentered = {200,500},
6929 <bch|blg|ugm> \textonesuperior = {200,300},
6930 <cmr|pad|pmn> \textonesuperior = {200,200},
6931 <ptm> \textonesuperior = {100,100},
6932 <bch|pad|pmn|ugm> \textordmasculine = {200,200},
6933 <blg|cmr> \textordmasculine = {100,200},
6934 <bch|cmr|pmn> \texteuro = {100, },
6935 <pad> \texteuro = { 50,100},
6936 <bch> \texttimes = {200,200},
6937 <blg|ptm> \texttimes = {100,100},
6938 <cmr> \texttimes = {150,250},
6939 <pad> \texttimes = {100,150},
6940 <pmn> \texttimes = { 70,100},
6941 <ugm> \texttimes = {200,300},
6942 <bch|pad|pmn> \textdiv = {150,200}
6943 <blg> \textdiv = {100,100}
6944 <cmr> \textdiv = {150,250}
6945 <ptm> \textdiv = { 50,100},
6946 <ugm> \textdiv = {200,300},
6947 <ptm> \textperthousand = { ,50}
6948 <ugm> \textsection = { ,100},
6949 <ugm> \textonehalf = { 50,100},
6950 <ugm> \textonequarter = { 50,100},
6951 <ugm> \textthreequarters = { 50,100},

```

```
6952 <ugm> \textsurd = { ,100}
```

Remaining slots in the source file.

```
6953 }
6954
6955 <*cmr|pad|pmn|ugm>
6956 \SetProtrusion
6957 <cmr> [ name = cmr-textcomp-it ]
6958 <pad> [ name = pad-textcomp-it ]
6959 <pmn> [ name = pmn-textcomp-it ]
6960 <ugm> [ name = ugm-textcomp-it ]
6961 { encoding = TS1,
6962 <cmr> family = cmr,
6963 <pad> family = {pad,padx,padj},
6964 <pmn> family = {pmnx,pmnj},
6965 <ugm> family = ugm,
6966 <!ugm> shape = {it,sl} }
6967 <ugm> shape = it }
6968 {
6969 <cmr> \textquotestraightbase = {300,600},
6970 <pad|pmn> \textquotestraightbase = {400,400},
6971 <cmr> \textquotestraightdblbase = {300,600},
6972 <pad> \textquotestraightdblbase = {300,400},
6973 <pmn> \textquotestraightdblbase = {300,300},
6974 \texttwelveudash = {200,200},
6975 <cmr|pad|pmn> \textthreequartersemdash = {150,150},
6976 <ugm> \textthreequartersemdash = {200,200},
6977 <cmr> \textquotesingle = {600,300},
6978 <pad> \textquotesingle = {800,100},
6979 <pmn> \textquotesingle = {300,200},
6980 <ugm> \textquotesingle = {500,500},
6981 <cmr> \textasteriskcentered = {300,200},
6982 <pad> \textasteriskcentered = {500,100},
6983 <pmn> \textasteriskcentered = {200,300},
6984 <ugm> \textasteriskcentered = {300,150},
6985 <pmn> \textfractionsolidus = {-200,-200},
6986 <cmr> \textoneoldstyle = {100, 50},
6987 <pad> \textoneoldstyle = {100, },
6988 <pmn> \textoneoldstyle = { 50, },
6989 <pad> \texttwooldstyle = { 50, },
6990 <pmn> \texttwooldstyle = {-50, },
6991 <cmr> \textthreeoldstyle = {100, 50},
6992 <pmn> \textthreeoldstyle = {-100, },
6993 <cmr> \textfouroldstyle = { 50, 50},
6994 <pad> \textfouroldstyle = { 50,100},
6995 <cmr> \textsevenoldstyle = { 50, 80},
6996 <pad> \textsevenoldstyle = { 50, },
6997 <pmn> \textsevenoldstyle = { 20, },
6998 <cmr> \textlangle = {400, },
6999 <cmr> \textrangle = { ,400},
7000 <cmr|pad> \textminus = {300,300},
7001 <pmn> \textminus = {200,200},
7002 <ugm> \textminus = {250,300},
7003 <pad|pmn> \textlbrackdbl = {100, },
7004 <pad|pmn> \textrbrackdbl = { ,100},
7005 <pmn> \textasciigrave = {300,300},
7006 <cmr|pad|pmn> \texttildelow = {200,250},
7007 <pmn> \textasciibreve = {300,300},
7008 <pmn> \textasciicaron = {300,300},
7009 <pmn> \textacutedbl = {200,300},
7010 <pmn> \textgravedbl = {150,300},
7011 <cmr> \textdagger = {100,100},
7012 <pad> \textdagger = {200,100},
7013 <pmn> \textdagger = { 80, 50},
7014 <ugm> \textdagger = { 80, 80},
```

```

7015 <cmr|pad> \textdaggerdbl = { 80, 80},
7016 <pmn> \textdaggerdbl = { 80, 50},
7017 <ugm> \textbardbl = {150,150},
7018 <cmr> \textbullet = {200,100},
7019 <pad> \textbullet = {300, },
7020 <pmn> \textbullet = { 30, 70},
7021 <ugm> \textbullet = { 50,100},
7022 <cmr> \textcelsius = {100, },
7023 <pad> \textcelsius = {200, },
7024 <pmn> \textcelsius = { 50,-50},
7025 <pad> \textflorin = {100, },
7026 <pmn> \textflorin = { 50,100},
7027 <ugm> \textflorin = { ,100},
7028 <cmr> \textcolonmonetary = {150, },
7029 <pad> \textcolonmonetary = {100, },
7030 <pmn> \textcolonmonetary = { 50,-50},
7031 <cmr|pad> \texttrademark = {200, },
7032 <pmn> \texttrademark = { 50,100},
7033 <ugm> \texttrademark = {150, 50},
7034 <ugm> \textcent = { 50, },
7035 <ugm> \textsterling = { , 50},
7036 <ugm> \textbrokenbar = {200,300},
7037 <pmn> \textasciidieresis = {300,200},
7038 <cmr> \textcopyright = {100, },
7039 <pad> \textcopyright = {200,100},
7040 <pmn> \textcopyright = {100,150},
7041 <ugm> \textcopyright = {300, },
7042 <cmr> \textordfeminine = {100,100},
7043 <pmn> \textordfeminine = {200,200},
7044 <ugm> \textordfeminine = {100,200},
7045 <cmr|pad> \textlnot = {300, },
7046 <pmn|ugm> \textlnot = {200, },
7047 <cmr> \textregistered = {100, },
7048 <pad> \textregistered = {200,100},
7049 <pmn> \textregistered = { 50,150},
7050 <ugm> \textregistered = {300, },
7051 <pmn> \textasciimacron = {150,200},
7052 <cmr|pad> \textdegree = {500,100},
7053 <pmn> \textdegree = {150,150},
7054 <ugm> \textdegree = {300,200},
7055 <cmr> \textpm = {150,100},
7056 <pad> \textpm = {200,150},
7057 <pmn|ugm> \textpm = {150,200},
7058 <cmr> \textonesuperior = {400, },
7059 <pad> \textonesuperior = {300,100},
7060 <pmn> \textonesuperior = {200,100},
7061 <ugm> \textonesuperior = {300,300},
7062 <cmr> \texttwosuperior = {400, },
7063 <pad> \texttwosuperior = {300, },
7064 <pmn> \texttwosuperior = {200,100},
7065 <ugm> \texttwosuperior = {300,200},
7066 <cmr> \textthreesuperior = {400, },
7067 <pad> \textthreesuperior = {300, },
7068 <pmn> \textthreesuperior = {200,100},
7069 <ugm> \textthreesuperior = {300,200},
7070 <ugm> \textmu = { ,100},
7071 <pmn> \textasciiacute = {300,200},
7072 <cmr> \textparagraph = {200, },
7073 <pmn> \textparagraph = { ,100},
7074 <cmr> \textperiodcentered = {500,500},
7075 <pad|pmn|ugm> \textperiodcentered = {300,400},
7076 <cmr> \textordmasculine = {100,100},
7077 <pmn> \textordmasculine = {200,200},
7078 <ugm> \textordmasculine = {300,200},
7079 <cmr> \texteuro = {200, },

```

```

7080 <pad> \texteuro = {100, },
7081 <pmn> \texteuro = {100,-50},
7082 <cmr> \texttimes = {200,200},
7083 <pad> \texttimes = {200,100},
7084 <pmn> \texttimes = { 70,100},
7085 <ugm> \texttimes = {200,300},
7086 <cmr|pad> \textdiv = {200,200}
7087 <pmn> \textdiv = {150,200}
7088 <ugm> \textdiv = {200,300},
7089 <ugm> \textsection = { ,200},
7090 <ugm> \textonehalf = { 50,100},
7091 <ugm> \textonequarter = { 50,100},
7092 <ugm> \textthreequarters = { 50,100},
7093 <ugm> \textsurd = { ,100}
7094 }
7095
7096 </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}

```

```

7097 <*cmr>
7098 \SetProtrusion
7099 [ name = cmr-math-letters ]
7100 { encoding = OML,
7101   family = cmm,
7102   series = {m,b},
7103   shape = it }
7104 {
7105   A = {100, 50}, % \mathnormal
7106   B = { 50, },
7107   C = { 50, },
7108   D = { 50, 50},
7109   E = { 50, },
7110   F = {100, 50},
7111   G = { 50, 50},
7112   H = { 50, 50},
7113   I = { 50, 50},
7114   J = {150, 50},
7115   K = { 50,100},
7116   L = { 50, 50},
7117   M = { 50, },
7118   N = { 50, },
7119   O = { 50, },
7120   P = { 50, },
7121   Q = { 50, 50},
7122   R = { 50, },
7123   S = { 50, },

```

```

7124     T = { 50,100},
7125     U = { 50, 50},
7126     V = {100,100},
7127     W = { 50,100},
7128     X = { 50,100},
7129     Y = {100,100},
7130     f = {100,100},
7131     h = {   ,100},
7132     i = {   , 50},
7133     j = {   , 50},
7134     k = {   , 50},
7135     r = {   , 50},
7136     v = {   , 50},
7137     w = {   , 50},
7138     x = {   , 50},
7139     "0B = { 50,100}, % \alpha
7140     "0C = { 50, 50}, % \beta
7141     "0D = {200,150}, % \gamma
7142     "0E = { 50, 50}, % \delta
7143     "0F = { 50, 50}, % \epsilon
7144     "10 = { 50,150}, % \zeta
7145     "12 = { 50,   }, % \theta
7146     "13 = {   ,100}, % \iota
7147     "14 = {   ,100}, % \kappa
7148     "15 = {100, 50}, % \lambda
7149     "16 = {   , 50}, % \mu
7150     "17 = {   , 50}, % \nu
7151     "18 = {   , 50}, % \xi
7152     "19 = { 50,100}, % \pi
7153     "1A = { 50, 50}, % \rho
7154     "1B = {   ,150}, % \sigma
7155     "1C = { 50,150}, % \tau
7156     "1D = { 50, 50}, % \upsilon
7157     "1F = { 50,100}, % \chi
7158     "20 = { 50, 50}, % \psi
7159     "21 = {   , 50}, % \omega
7160     "22 = {   , 50}, % \varepsilon
7161     "23 = {   , 50}, % \vartheta
7162     "24 = {   , 50}, % \varpi
7163     "25 = {100,   }, % \varrho
7164     "26 = {100,100}, % \varsigma
7165     "27 = { 50, 50}, % \varphi
7166     "28 = {100,100}, % \leftharpoonup
7167     "29 = {100,100}, % \leftharpoondown
7168     "2A = {100,100}, % \rightharpoonup
7169     "2B = {100,100}, % \rightharpoondown
7170     "2C = {300,200}, % \lhook
7171     "2D = {200,300}, % \rhook
7172     "2E = {   ,100}, % \triangleright
7173     "2F = {100,   }, % \triangleleft
7174     "3A = {   ,500}, % ., \ldotp
7175     "3B = {   ,500}, % ,
7176     "3C = {200,100}, % <
7177     "3D = {300,400}, % /
7178     "3E = {100,200}, % >
7179     "3F = {200,200}, % \star
7180     "5B = {   ,100}, % \flat
7181     "5E = {200,200}, % \smile
7182     "5F = {200,200}, % \frown
7183     "7C = {100,   }, % \jmath
7184     "7D = {   ,100} % \wp

```

Remaining slots in the source file.

```

7185     }
7186

```

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}
```

```
7187 \SetProtrusion
7188 [ name = cmr-math-symbols ]
7189 { encoding = OMS,
7190   family = cmsy,
7191   series = {m,b},
7192   shape = n }
7193 {
7194   A = {150, 50}, % \mathcal
7195   C = { ,100},
7196   D = { , 50},
7197   F = { 50,150},
7198   I = { ,100},
7199   J = {100,150},
7200   K = { ,100},
7201   L = {100, },
7202   M = { 50, 50},
7203   N = { 50,100},
7204   P = { , 50},
7205   Q = { 50, },
7206   R = { , 50},
7207   T = { 50,150},
7208   V = { 50, 50},
7209   W = { , 50},
7210   X = {100,100},
7211   Y = {100, },
7212   Z = {100,150},
7213   "00 = {300,300}, % -
7214   "01 = { ,700}, % \cdot, \cdotp
7215   "02 = {150,250}, % \times
7216   "03 = {150,250}, % *, \ast
7217   "04 = {200,300}, % \div
7218   "05 = {150,250}, % \diamond
7219   "06 = {200,200}, % \pm
7220   "07 = {200,200}, % \mp
7221   "08 = {100,100}, % \oplus
7222   "09 = {100,100}, % \ominus
7223   "0A = {100,100}, % \otimes
7224   "0B = {100,100}, % \oslash
7225   "0C = {100,100}, % \odot
7226   "0D = {100,100}, % \bigcirc
7227   "0E = {100,100}, % \circ
7228   "0F = {100,100}, % \bullet
7229   "10 = {100,100}, % \asymp
7230   "11 = {100,100}, % \equiv
7231   "12 = {200,100}, % \subseteq
7232   "13 = {100,200}, % \supseteq
7233   "14 = {200,100}, % \leq
7234   "15 = {100,200}, % \geq
7235   "16 = {200,100}, % \preceq
7236   "17 = {100,200}, % \succeq
7237   "18 = {200,200}, % \sim
7238   "19 = {150,150}, % \approx
7239   "1A = {200,100}, % \subset
7240   "1B = {100,200}, % \supset
7241   "1C = {200,100}, % \ll
7242   "1D = {100,200}, % \gg
7243   "1E = {300,100}, % \prec
7244   "1F = {100,300}, % \succ
7245   "20 = {100,200}, % \leftarrow
7246   "21 = {200,100}, % \rightarrow
7247   "22 = {100,100}, % \uparrow
```



```

7248 "23 = {100,100}, % \downarrow
7249 "24 = {100,100}, % \leftrightarrows
7250 "25 = {100,100}, % \nearrow
7251 "26 = {100,100}, % \searrow
7252 "27 = {100,100}, % \simeq
7253 "28 = {100,100}, % \Leftarrow
7254 "29 = {100,100}, % \Rightarrow
7255 "2A = {100,100}, % \Uparrow
7256 "2B = {100,100}, % \Downarrow
7257 "2C = {100,100}, % \Leftrightarrow
7258 "2D = {100,100}, % \nrightarrow
7259 "2E = {100,100}, % \swarrow
7260 "2F = { ,100}, % \propto
7261 "30 = { ,400}, % \prime
7262 "31 = {100,100}, % \infty
7263 "32 = {150,100}, % \in
7264 "33 = {100,150}, % \ni
7265 "34 = {100,100}, % \triangle, \bigtriangleup
7266 "35 = {100,100}, % \bigtriangledown
7267 "38 = { ,100}, % \forall
7268 "39 = {100, }, % \exists
7269 "3A = {200, }, % \neg
7270 "3E = {200,200}, % \top
7271 "3F = {200,200}, % \bot, \perp
7272 "5E = {100,200}, % \wedge
7273 "5F = {100,200}, % \vee
7274 "60 = { ,300}, % \vdash
7275 "61 = {300, }, % \dashv
7276 "62 = {100,100}, % \lfloor
7277 "63 = {100,100}, % \rfloor
7278 "64 = {100,100}, % \lceil
7279 "65 = {100,100}, % \rceil
7280 "66 = {150, }, % \lbrace
7281 "67 = { ,150}, % \rbrace
7282 "68 = {400, }, % \langle
7283 "69 = { ,400}, % \rangle
7284 "6C = {100,100}, % \updownarrow
7285 "6D = {100,100}, % \Updownarrow
7286 "6E = {100,300}, % \, \backslash, \setminus
7287 "72 = {100,100}, % \nabla
7288 "79 = {200,200}, % \dagger
7289 "7A = {100,100}, % \ddagger
7290 "7B = {100, }, % \mathparagraph
7291 "7C = {100,100}, % \clubsuit
7292 "7D = {100,100}, % \diamondsuit
7293 "7E = {100,100}, % \heartsuit
7294 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

7295 }
7296

```

We don't bother about 'largsymbols', since it will only be used in display math, where protrusion doesn't work anyway. It's declared as:

```
\DeclareSymbolFont{largsymbols}{OMX}{cmex}{m}{n}
```

```

7297 </cmr>
7298 </cfg-t>

```

15.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
7299 <*cfg-u>
```

Symbol font 'a'.

```

7300 (*msa)
7301 \SetProtrusion
7302 [ name = AMS-a ]
7303 { encoding = U,
7304   family = msa }
7305 {
7306   "05 = {150,250}, % \centerdot
7307   "06 = {100,100}, % \lozenge
7308   "07 = { 50, 50}, % \blacklozenge
7309   "08 = { 50, 50}, % \circlearrowright
7310   "09 = { 50, 50}, % \circlearrowleft
7311   "0A = {100,100}, % \rightleftharpoons
7312   "0B = {100,100}, % \leftrightharpoons
7313   "0D = {-50,200}, % \Vdash
7314   "0E = {-50,200}, % \Vvdash
7315   "0F = {-70,150}, % \vDash
7316   "10 = {100,150}, % \twoheadrightarrow
7317   "11 = {100,150}, % \twoheadleftarrow
7318   "12 = { 50,100}, % \leftleftarrows
7319   "13 = { 50, 80}, % \rightrightarrows
7320   "14 = {120,120}, % \upuparrows
7321   "15 = {120,120}, % \downdownarrows
7322   "16 = {200,200}, % \upharpoonright
7323   "17 = {200,200}, % \downharpoonright
7324   "18 = {200,200}, % \upharpoonleft
7325   "19 = {200,200}, % \downharpoonleft
7326   "1A = { 80,100}, % \rightarrowtail
7327   "1B = { 80,100}, % \leftarrowtail
7328   "1C = { 50, 50}, % \leftrightarrows
7329   "1D = { 50, 50}, % \rightleftarrows
7330   "1E = {250,  }, % \Lsh
7331   "1F = {  ,250}, % \Rsh
7332   "20 = {100,100}, % \rightsquigarrow
7333   "21 = {100,100}, % \leftrightsquigarrow
7334   "22 = {100, 50}, % \looparrowleft
7335   "23 = { 50,100}, % \looparrowright
7336   "24 = { 50, 80}, % \circeq
7337   "25 = {  ,100}, % \succsim
7338   "26 = {  ,100}, % \gtrsim
7339   "27 = {  ,100}, % \gtrapprox
7340   "28 = {150, 50}, % \multimap
7341   "2B = {100,150}, % \doteqdot
7342   "2C = {100,150}, % \triangleq
7343   "2D = {100, 50}, % \precsim
7344   "2E = {100, 50}, % \lessim
7345   "2F = { 50, 50}, % \lessapprox
7346   "30 = {100, 50}, % \eqslantless
7347   "31 = { 50, 50}, % \eqslantgtr
7348   "32 = {100, 50}, % \curlyeqprec
7349   "33 = { 50,100}, % \curlyeqsucc
7350   "34 = {100, 50}, % \preccurlyeq
7351   "36 = { 50,  }, % \leqslant
7352   "38 = {  , 50}, % \backprime
7353   "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
7354   "3C = { 50,100}, % \succcurlyeq
7355   "3E = {  , 50}, % \geqslant
7356   "40 = {  , 50}, % \sqsubset
7357   "41 = { 50,  }, % \sqsupset
7358   "42 = {  ,150}, % \vartriangleright, \rhd
7359   "43 = {150,  }, % \vartriangleleft, \lhd
7360   "44 = {  ,100}, % \trianglerighteq, \unrhd
7361   "45 = {100,  }, % \trianglelefteq, \unlhd
7362   "46 = {100,100}, % \bigstar
7363   "48 = { 50, 50}, % \blacktriangledown

```

```

7364 "49 = { ,100}, % \blacktriangleright
7365 "4A = {100, }, % \blacktriangleleft
7366 "4B = { ,150}, % \dashrightarrow (the arrow)
7367 "4C = {150, }, % \dashleftarrow
7368 "4D = { 50, 50}, % \vartriangle
7369 "4E = { 50, 50}, % \blacktriangle
7370 "4F = { 50, 50}, % \triangledown
7371 "50 = { 50, 50}, % \eqcirc
7372 "56 = { ,150}, % \Rrightarrow
7373 "57 = {150, }, % \Lleftarrow
7374 "58 = {100,300}, % \checkmark
7375 "5C = { 50, 50}, % \angle
7376 "5D = { 50, 50}, % \measuredangle
7377 "5E = { 50, 50}, % \sphericalangle
7378 "5F = { , 50}, % \varpropto
7379 "60 = {100,100}, % \smallsmile
7380 "61 = {100,100}, % \smallfrown
7381 "62 = { 50, }, % \Subset
7382 "63 = { , 50}, % \Supset
7383 "66 = {150,150}, % \curlywedge
7384 "67 = {150,150}, % \curlyvee
7385 "68 = { 50,150}, % \leftthreetimes
7386 "69 = {100, 50}, % \rightthreetimes
7387 "6C = { 50, 50}, % \bumpeq
7388 "6D = { 50, 50}, % \Bumpeq
7389 "6E = {100, }, % \lll
7390 "6F = { ,100}, % \ggg
7391 "70 = { 50,100}, % \ulcorner
7392 "71 = {100, 50}, % \urcorner
7393 "75 = {150,200}, % \dotplus
7394 "76 = { 50,100}, % \backsimeq
7395 "78 = { 50,100}, % \llcorner
7396 "79 = {100, 50}, % \lrcorner
7397 "7C = {100,100}, % \intercal
7398 "7D = { 50, 50}, % \circledcirc
7399 "7E = { 50, 50}, % \circledast
7400 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

7401 }
7402
7403 </msa>

```

Symbol font 'b'.

```

7404 <+msb>
7405 \SetProtrusion
7406 [ name = AMS-b ]
7407 { encoding = U,
7408 family = msb }
7409 {
7410 A = { 50, 50}, % \mathbb
7411 C = { 50, 50},
7412 G = { , 50},
7413 L = { , 50},
7414 P = { , 50},
7415 R = { , 50},
7416 T = { , 50},
7417 V = { 50, 50},
7418 X = { 50, 50},
7419 Y = { 50, 50},
7420 "00 = { 50, 50}, % \lvertneqq
7421 "01 = { 50, 50}, % \gvertneqq
7422 "02 = { 50, 50}, % \nleq
7423 "03 = { 50, 50}, % \ngeq
7424 "04 = {100, 50}, % \nless

```

```

7425 "05 = { 50,150}, % \ngtr
7426 "06 = {100, 50}, % \nprec
7427 "07 = { 50,150}, % \nsucc
7428 "08 = { 50, 50}, % \lneqq
7429 "09 = { 50, 50}, % \gneqq
7430 "0A = {100,100}, % \lneqslant
7431 "0B = {100,100}, % \ngeqslant
7432 "0C = {100, 50}, % \lneq
7433 "0D = { 50,100}, % \gneq
7434 "0E = {100, 50}, % \npreceq
7435 "0F = { 50,100}, % \nsucceq
7436 "10 = { 50,  }, % \precnsim
7437 "11 = { 50, 50}, % \succnsim
7438 "12 = { 50, 50}, % \lnsim
7439 "13 = { 50, 50}, % \gnsim
7440 "14 = { 50, 50}, % \lneqq
7441 "15 = { 50, 50}, % \gneqq
7442 "16 = { 50, 50}, % \precneqq
7443 "17 = { 50, 50}, % \succneqq
7444 "18 = { 50, 50}, % \precnapprox
7445 "19 = { 50, 50}, % \succnapprox
7446 "1A = { 50, 50}, % \lnapprox
7447 "1B = { 50, 50}, % \gnapprox
7448 "1C = {150,200}, % \nsim
7449 "1D = { 50, 50}, % \ncong
7450 "1E = {100,150}, % \diagup
7451 "1F = {100,150}, % \diagdown
7452 "20 = {100, 50}, % \varsubsetneq
7453 "21 = { 50,100}, % \varsupsetneq
7454 "22 = {100, 50}, % \subsetneqq
7455 "23 = { 50,100}, % \supsetneqq
7456 "24 = {100, 50}, % \subsetneqq
7457 "25 = { 50,100}, % \supsetneqq
7458 "26 = {100, 50}, % \varsubsetneqq
7459 "27 = { 50,100}, % \varsupsetneqq
7460 "28 = {100, 50}, % \subsetneq
7461 "29 = { 50,100}, % \supsetneq
7462 "2A = {100, 50}, % \subseteq
7463 "2B = { 50,100}, % \supseteq
7464 "2C = { 50,100}, % \nparallel
7465 "2D = {100,150}, % \nmid
7466 "2E = {150,150}, % \nshortmid
7467 "2F = {100,100}, % \nshortparallel
7468 "30 = {  ,150}, % \nvDash
7469 "31 = {  ,150}, % \nVDash
7470 "32 = {  ,100}, % \nvDash
7471 "33 = {  ,100}, % \nVDash
7472 "34 = {  ,100}, % \ntrianglerighteq
7473 "35 = {100,  }, % \trianglelefteq
7474 "36 = {100,  }, % \triangleleft
7475 "37 = {  ,100}, % \triangleright
7476 "38 = {100,200}, % \leftarrow
7477 "39 = {100,200}, % \rightarrow
7478 "3A = {100,100}, % \Leftarrow
7479 "3B = { 50,100}, % \Rightarrow
7480 "3C = {100,100}, % \Leftrightarrow
7481 "3D = {100,200}, % \leftrightharrow
7482 "3E = { 50, 50}, % \divideontimes
7483 "3F = { 50, 50}, % \varnothing
7484 "60 = {200,  }, % \Finv
7485 "61 = {  , 50}, % \Game
7486 "68 = {100,100}, % \eqsim
7487 "69 = { 50,  }, % \beth
7488 "6A = { 50,  }, % \gimel
7489 "6B = {150,  }, % \daleth

```

```

7490 "6C = {200, }, % \lessdot
7491 "6D = { ,200}, % \gtrdot
7492 "6E = {100,200}, % \ltimes
7493 "6F = {150,100}, % \rtimes
7494 "70 = { 50,100}, % \shortmid
7495 "71 = { 50, 50}, % \shortparallel
7496 "72 = {200,300}, % \smallsetminus
7497 "73 = {100,200}, % \thicksim
7498 "74 = { 50,100}, % \thickapprox
7499 "75 = { 50, 50}, % \approxeq
7500 "76 = { 50,100}, % \succapprox
7501 "77 = { 50, 50}, % \precapprox
7502 "78 = {100,100}, % \curvearrowleft
7503 "79 = { 50,150}, % \curvearrowright
7504 "7A = { 50,200}, % \digamma
7505 "7B = {100, 50}, % \varkappa
7506 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

7507 }
7508
7509 (msb)

```

15.8.8 Euler

Euler Roman font (package `euler`).

```

7510 (*eur)
7511 \SetProtrusion
7512 [ name = euler ]
7513 { encoding = U,
7514   family = eur }
7515 {
7516 "01 = {100,100},
7517 "03 = {100,150},
7518 "06 = { ,100},
7519 "07 = {100,150},
7520 "08 = {100,100},
7521 "0A = {100,100},
7522 "0B = { , 50},
7523 "0C = { ,100},
7524 "0D = {100,100},
7525 "0E = { ,100},
7526 "0F = {100,100},
7527 "10 = {100,100},
7528 "13 = { ,100},
7529 "14 = { ,100},
7530 "15 = { , 50},
7531 "16 = { , 50},
7532 "17 = { 50,100},
7533 "18 = { 50,100},
7534 "1A = { , 50},
7535 "1B = { , 50},
7536 "1C = { 50,100},
7537 "1D = { 50,100},
7538 "1E = { 50,100},
7539 "1F = { 50,100},
7540 "20 = { , 50},
7541 "21 = { , 50},
7542 "22 = { 50,100},
7543 "24 = { , 50},
7544 "27 = { 50,100},
7545 1 = {100,100},
7546 7 = { 50,100},
7547 "3A = {300,500},

```

```

7548     "3B = {200,400},
7549     "3C = {200,100},
7550     "3D = {200,200},
7551     "3E = {100,200},
7552     A = { ,100},
7553     D = { , 50},
7554     J = { 50, },
7555     K = { , 50},
7556     L = { , 50},
7557     Q = { , 50},
7558     T = { 50, },
7559     X = { 50, 50},
7560     Y = { 50, },
7561     h = { , 50},
7562     k = { , 50}
7563   }
7564

```

Extended by the `eulerm` package.

```

7565 \SetProtrusion
7566   [ name = euler-vm,
7567     load = euler ]
7568   { encoding = U,
7569     family = zeur }
7570   {
7571     "28 = {100,200},
7572     "29 = {100,200},
7573     "2A = {100,150},
7574     "2B = {100,150},
7575     "2C = {200,300},
7576     "2D = {200,300},
7577     "2E = { ,100},
7578     "2F = {100, },
7579     "3F = {150,150},
7580     "5B = { ,100},
7581     "5E = {100,100},
7582     "5F = {100,100},
7583     "80 = { , 50},
7584     "81 = {200,250},
7585     "82 = {100,200}
7586   }
7587
7588 (/eur)

```

Euler Script font (`euca1`).

```

7589 (*eus)
7590 \SetProtrusion
7591   [ name = euscript ]
7592   { encoding = U,
7593     family = eus }
7594   {
7595     A = {100,100},
7596     B = { 50,100},
7597     C = { 50, 50},
7598     D = { 50,100},
7599     E = { 50,100},
7600     F = { 50, },
7601     G = { 50, },
7602     H = { ,100},
7603     K = { , 50},
7604     L = { ,150},
7605     M = { , 50},
7606     N = { , 50},
7607     O = { 50, 50},
7608     P = { 50, 50},

```

```
7609     T = {   ,100},
7610     U = {   , 50},
7611     V = { 50, 50},
7612     W = { 50, 50},
7613     X = { 50, 50},
7614     Y = { 50,   },
7615     Z = { 50,100},
7616     "00 = {250,250},
7617     "18 = {200,200},
7618     "3A = {200,150},
7619     "40 = {   ,100},
7620     "5E = {100,100},
7621     "5F = {100,100},
7622     "66 = { 50,   },
7623     "67 = {   , 50},
7624     "6E = {200,200}
7625   }
7626
7627 \SetProtrusion
7628   [ name   = euscript-vm,
7629     load   = euscript ]
7630   { encoding = U,
7631     family  = zeus }
7632   {
7633     "01 = {600,600},
7634     "02 = {200,200},
7635     "03 = {200,200},
7636     "04 = {200,200},
7637     "05 = {150,150},
7638     "06 = {200,200},
7639     "07 = {200,200},
7640     "08 = {100,100},
7641     "09 = {100,100},
7642     "0A = {100,100},
7643     "0B = {100,100},
7644     "0C = {100,100},
7645     "0D = {100,100},
7646     "0E = {150,150},
7647     "0F = {100,100},
7648     "10 = {150,150},
7649     "11 = {100,100},
7650     "12 = {150,100},
7651     "13 = {100,150},
7652     "14 = {150,100},
7653     "15 = {100,150},
7654     "16 = {200,100},
7655     "17 = {100,200},
7656     "19 = {150,150},
7657     "1A = {150,100},
7658     "1B = {100,150},
7659     "1C = {100,100},
7660     "1D = {100,100},
7661     "1E = {250,100},
7662     "1F = {100,250},
7663     "20 = {150,200},
7664     "21 = {150,200},
7665     "22 = {150,150},
7666     "23 = {150,150},
7667     "24 = {100,200},
7668     "25 = {150,150},
7669     "26 = {150,150},
7670     "27 = {100,100},
7671     "28 = {100,100},
7672     "29 = {100,150},
7673     "2A = {100,100},
```

```

7674 "2B = {100,100},
7675 "2C = {100,100},
7676 "2D = {150,150},
7677 "2E = {150,150},
7678 "2F = {100,100},
7679 "30 = {100,100},
7680 "31 = {100,100},
7681 "32 = {100,100},
7682 "33 = {100,100},
7683 "34 = {100,100},
7684 "35 = {100,100},
7685 "3E = {150,150},
7686 "3F = {150,150},
7687 "60 = { ,200},
7688 "61 = {200, },
7689 "62 = {100,100},
7690 "63 = {100,100},
7691 "64 = {100,100},
7692 "65 = {100,100},
7693 "68 = {300, },
7694 "69 = { ,300},
7695 "6C = {100,100},
7696 "6D = {100,100},
7697 "6F = {100,100},
7698 "72 = {100,100},
7699 "73 = {200,100},
7700 "76 = { ,100},
7701 "77 = {100, },
7702 "78 = { 50, 50},
7703 "79 = {100,100},
7704 "7A = {100,100},
7705 "7D = {150,150},
7706 "7E = {100,100},
7707 "A8 = {100,100},
7708 "A9 = {100,100},
7709 "AB = {200,200},
7710 "BA = { ,200},
7711 "BB = { ,200},
7712 "BD = {200,200},
7713 "DE = {200,200}
7714 }
7715
7716 (<eus)

```

Euler Fraktur font (eufrak).

```

7717 (<*euf)
7718 \SetProtrusion
7719 [ name = mathfrak ]
7720 { encoding = U,
7721 family = euf }
7722 {
7723 A = { , 50},
7724 B = { , 50},
7725 C = { 50, 50},
7726 D = { , 80},
7727 E = { 50, },
7728 G = { , 50},
7729 L = { , 80},
7730 O = { , 50},
7731 T = { , 80},
7732 X = { 80, 50},
7733 Z = { 80, 50},
7734 b = { , 50},
7735 c = { , 50},
7736 k = { , 50},

```



```

7737     p = {   , 50},
7738     q = { 50,   },
7739     v = {   , 50},
7740     w = {   , 50},
7741     x = {   , 50},
7742     1 = {100,100},
7743     2 = { 80, 80},
7744     3 = { 80, 50},
7745     4 = { 80, 50},
7746     7 = { 50, 50},
7747     "12 = {500,500},
7748     "13 = {500,500},
7749     ! = {   ,200},
7750     ' = {200,300},
7751     ( = {200,   },
7752     ) = {   ,200},
7753     * = {200,200},
7754     + = {200,250},
7755     - = {200,200},
7756     {,} = {300,300},
7757     . = {400,400},
7758     {=} = {200,200},
7759     : = {   ,200},
7760     ; = {   ,200},
7761     ] = {   ,200}
7762   }
7763
7764 </euf>
7765 </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).

```

7766 <+cfg-e>
7767 \SetProtrusion
7768 <zpeu|euroitc> { encoding = U,
7769 <mvs> { encoding = {OT1,U},
7770 <zpeu> family = zpeu }
7771 <euroitc> family = {euroitc,euroitcs} }
7772 <mvs> family = mvs }
7773 {
7774 <zpeu> E = {50, }
7775 <euroitc> E = {100,50}
7776 <mvs> 164 = {50,50}, % \EUR
7777 <mvs> 068 = {50,-100} % \EURdig
7778 }
7779
7780 <+zpeu|euroitc>
7781 \SetProtrusion
7782 { encoding = U,
7783 <zpeu> family = zpeu,
7784 <euroitc> family = {euroitc,euroitcs},
7785 shape = it* }
7786 {
7787 <zpeu> E = {100,-50}
7788 <euroitc> E = {100,}
7789 }
7790
7791 </zpeu|euroitc>
7792 <+zpeu>
7793 \SetProtrusion

```

24 Of course, there are many more symbols in this font. Feel free to contribute protrusion settings!

Figure 1:

Example of interword spacing (from: M. Siemoneit, *Typographisches Gestalten*, Frankfurt/M. 1989). The numbers indicate the preference for shrinking the interword space.

2 6 7 5 3 4 1

Das Aus kam in der letzten Runde, wobei

Das Aus kam in der letzten Runde, wobei

Das Aus kam in der letzten Runde, wobei

Das Aus kam in der letzten Runde, wobei

Das Aus kam in der letzten Runde, wobei

```

7794 { encoding = U,
7795     family   = {zpeus,eurosans} }
7796 {
7797     E = {100,50}
7798 }
7799
7800 \SetProtrusion
7801 { encoding = U,
7802     family   = {zpeus,eurosans},
7803     shape    = it* }
7804 {
7805     E = {200, }
7806 }
7807
7808 </zpeu>
7809 </cfg-e>

```

15.9 Interword spacing

Default unit is space.

```

7810 <*m-t|cmr>
7811 %%% -----
7812 %%% INTERWORD SPACING
7813
7814 </m-t|cmr>
7815 <*m-t>
7816 \SetExtraSpacing
7817 [ name = default ]
7818 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
7819 {

```

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

```
7820 { , } = { , -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)

7821 r = { , -300, 300 },

- [before or] after lowercase characters with ascenders

7822 b = { , -200, 200 },

7823 d = { , -200, 200 },

7824 f = { , -200, 200 },

7825 h = { , -200, 200 },

7826 k = { , -200, 200 },

7827 l = { , -200, 200 },

7828 t = { , -200, 200 },

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’

7829 c = { , -100, 100 },

7830 p = { , -100, 100 },

7831 v = { , -100, 100 },

7832 w = { , -100, 100 },

7833 z = { , -100, 100 },

7834 x = { , -100, 100 },

7835 y = { , -100, 100 },

- [before or] after lowercase characters with x-height plus descender without additional optical space

7836 i = { , 50, -50 },

7837 m = { , 50, -50 },

7838 n = { , 50, -50 },

7839 u = { , 50, -50 },

- after colon and semicolon

7840 : = { , 200, -200 },

7841 ; = { , 200, -200 },

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

7842 . = { , 250, -250 },

7843 ! = { , 250, -250 },

7844 ? = { , 250, -250 }

The order has to be reversed when enlarging is needed.’

7845 }

7846

7847 *(/m-t)*

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000.)
- Should the first parameter also be used? (Probably.)
- What about quotation marks, parentheses etc.?

Furthermore, there seems to be a pdfTeX bug with spacing in combination with a non-zero \spaceskip (reported by *Axel Berger*):

```
\parfillskip0pt
\rightskip0pt plus 1em
\spaceskip\fontdimen2\font
test test\par
\pdfadjustinterwordglue2
```

```
\stbrcode\font`t=-50
test test
\bye
```

Some more characters in T2A.²⁵

```
7848 (*cmr)
7849 \SetExtraSpacing
7850   [ name      = T2A,
7851     load      = default ]
7852   { encoding = T2A,
7853     family   = cmr }
7854   {
7855     \cyrg = { , -300, 300},
7856     \cyrb = { , -200, 200},
7857     \cyrk = { , -200, 200},
7858     \cyrs = { , -100, 100},
7859     \cyrr = { , -100, 100},
7860     \cyrh = { , -100, 100},
7861     \cyru = { , -100, 100},
7862     \cyrt = { , 50, -50},
7863     \cyrp = { , 50, -50},
7864     \cyri = { , 50, -50},
7865     \cyrishrt = { , 50, -50},
7866   }
7867
```

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.

```
7868 \SetExtraSpacing
7869   [ name      = nonfrench-cmr,
7870     load      = default,
7871     context   = nonfrench ]
7872   { encoding = {OT1,T1,LY1,OT4,QX,T5},
7873     family   = cmr }
7874   {
```

`latex.ltx` has:

```
\def\nonfrenchspacing{
\sffcode`. 3000
\sffcode`? 3000
\sffcode`! 3000
```

```
7875   . = {333,2000,-667},
7876   ? = {333,2000,-667},
7877   ! = {333,2000,-667},
```

```
\sffcode`\: 2000
```

²⁵ Contributed by *Karl Karlsson*.

```
7878 : = {333,1000,-500},
```

```
\sfcode`\; 1500
```

```
7879 ; = { , 500,-333},
```

```
\sfcode`\, 1250
```

```
7880 {,}= { , 250,-200}
```

```
}
```

```
7881 }
```

```
7882
```

```
7883 /cmr
```

fontinst, however, which is also used to create the psnfss font metrics, sets \fontdimen 7 to 240 by default. Therefore, the fallback settings use this value for the first component.

```
7884 *m-t
```

```
7885 \SetExtraSpacing
```

```
7886 [ name = nonfrench-default,
```

```
7887 load = default,
```

```
7888 context = nonfrench ]
```

```
7889 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
```

```
7890 {
```

```
7891 . = {240,2000,-667},
```

```
7892 ? = {240,2000,-667},
```

```
7893 ! = {240,2000,-667},
```

```
7894 : = {240,1000,-500},
```

```
7895 ; = { , 500,-333},
```

```
7896 {,}= { , 250,-200}
```

```
7897 }
```

```
7898
```

15.10 Additional kerning

Default unit is 1 em.

```
7899 %%% -----
```

```
7900 %%% ADDITIONAL KERNING
```

```
7901
```

A dummy list to be loaded when no context is active.

```
7902 \SetExtraKerning
```

```
7903 [ name = empty ]
```

```
7904 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
```

```
7905 { }
```

```
7906
```

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.)

```

7907 \SetExtraKerning
7908   [ name      = french-default,
7909     context   = french,
7910     unit      = space ]
7911   { encoding = {OT1,T1,LY1} }
7912   {
7913     : = {1000,}, % = \fontdimen2
7914     ; = {500, }, % ~ \thinspace
7915     ! = {500, },
7916     ? = {500, }
7917   }
7918

```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```

7919 \SetExtraKerning
7920   [ name      = french-guillemets,
7921     context   = french-guillemets,
7922     load      = french-default,
7923     unit      = space ]
7924   { encoding = {T1,LY1} }
7925   {
7926     \guillemotleft = { ,800}, % = 0.8\fontdimen2
7927     \guillemotright = {800, }
7928   }
7929
7930 \SetExtraKerning
7931   [ name      = french-guillemets-OT1,
7932     context   = french-guillemets,
7933     load      = french-default,
7934     unit      = space ]
7935   { encoding = OT1 }
7936   { }
7937

```

15.10.2 Turkish

```

7938 \SetExtraKerning
7939   [ name      = turkish,
7940     context   = turkish ]
7941   { encoding = {OT1,T1,LY1} }
7942   {
7943     : = {167, }, % = \thinspace
7944     ! = {167, },
7945     {=} = {167, }
7946   }
7947
7948 </m-t>
7949 </config>

```

26 http://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

8055 T = {T, T̄, T̂, T̃, T̄, T̅, T̆, Ṫ, T̈, T̉, T̊, T̋, Ť, T̍, T̎, T̏, T̐, T̑, T̒, T̓, T̔, T̕, T̖, T̗, T̘, T̙, T̚, T̛, T̜, T̝, T̞, T̟, T̠, T̡, T̢, Ṭ, T̤, T̥, Ț, Ţ, T̨, T̩, T̪, T̫, T̬, Ṱ, T̮, T̯, T̰, Ṯ, T̲, T̳, T̴, T̵, T̶, T̷, T̸, T̹, T̺, T̻, T̼, T̽, T̾, T̿, T̿̄, T̿̅, T̿̆, T̿̇, T̿̈, T̿̉, T̿̊, T̿̋, T̿̌, T̿̍, T̿̎, T̿̏, T̿̐, T̿̑, T̿̒, T̿̓, T̿̔, T̿̕, T̖̿, T̗̿, T̘̿, T̙̿, T̿̚, T̛̿, T̜̿, T̝̿, T̞̿, T̟̿, T̠̿, T̡̿, T̢̿, Ṭ̿, T̤̿, T̥̿, Ț̿, Ţ̿, T̨̿, T̩̿, T̪̿, T̫̿, T̬̿, Ṱ̿, T̮̿, T̯̿, T̰̿, Ṯ̿, T̲̿, T̳̿, T̴̿, T̵̿, T̶̿, T̷̿, T̸̿, T̹̿, T̺̿, T̻̿, T̼̿, T̿̽, T̿̾, T̿̿}, % Cyr
8056 T, T̄}, % Cyr
8057 U = {U, Ū, Ū̄, Ū̂, Ū̃, Ū̄, Ū̅, Ū̆, Ū̇, Ṻ, Ū̉, Ū̊, Ū̋, Ū̌, Ū̍, Ū̎, Ū̏, Ū̐, Ū̑, Ū̒, Ū̓, Ū̔, Ū̕, Ū̖, Ū̗, Ū̘, Ū̙, Ū̚, Ư̄, Ū̜, Ū̝, Ū̞, Ū̟, Ū̠, Ū̡, Ū̢, Ụ̄, Ṳ̄, Ū̥, Ū̦, Ū̧, Ų̄, Ū̩, Ū̪, Ū̫, Ū̬, Ṷ̄, Ū̮, Ū̯, Ṵ̄, Ū̱, Ū̲, Ū̳, Ū̴, Ū̵, Ū̶, Ū̷, Ū̸, Ū̹, Ū̺, Ū̻, Ū̼, Ū̽, Ū̾, Ū̿}, % Cyr
8058 V = {V, V̄},
8059 W = {W̄, Ŵ, W̃, W̄, W̅, W̆, Ẇ, Ẅ, W̉, W̊, W̋, W̌, W̍, W̎, W̏, W̐, W̑, W̒, W̓, W̔, W̕, W̖, W̗, W̘, W̙, W̚, W̛, W̜, W̝, W̞, W̟, W̠, W̡, W̢, Ẉ, W̤, W̥, W̦, W̧, W̨, W̩, W̪, W̫, W̬, W̭, W̮, W̯, W̰, W̱, W̲, W̳, W̴, W̵, W̶, W̷, W̸, W̹, W̺, W̻, W̼, W̽, W̾, W̿}, % Cyr
8060 W}, % Cyr
8061 X = {X, X̄, X̂, X̃, X̄, X̅, X̆, Ẋ, Ẍ, X̉, X̊, X̋, X̌, X̍, X̎, X̏, X̐, X̑, X̒, X̓, X̔, X̕, X̖, X̗, X̘, X̙, X̚, X̛, X̜, X̝, X̞, X̟, X̠, X̡, X̢, X̣, X̤, X̥, X̦, X̧, X̨, X̩, X̪, X̫, X̬, X̭, X̮, X̯, X̰, X̱, X̲, X̳, X̴, X̵, X̶, X̷, X̸, X̹, X̺, X̻, X̼, X̽, X̾, X̿}, % Cyr
8062 X, X̄, X̂, X̃}, % Cyr
8063 Y = {Y, Ÿ, Ÿ̄, Ÿ̂, Ÿ̃, Ÿ̄, Ÿ̅, Ÿ̆, Ÿ̇, Ÿ̈, Ÿ̉, Ÿ̊, Ÿ̋, Ÿ̌, Ÿ̍, Ÿ̎, Ÿ̏, Ÿ̐, Ÿ̑, Ÿ̒, Ÿ̓, Ÿ̔, Ÿ̕, Ÿ̖, Ÿ̗, Ÿ̘, Ÿ̙, Ÿ̚, Ÿ̛, Ÿ̜, Ÿ̝, Ÿ̞, Ÿ̟, Ÿ̠, Ÿ̡, Ÿ̢, Ỵ̈, Ÿ̤, Ÿ̥, Ÿ̦, Ÿ̧, Ÿ̨, Ÿ̩, Ÿ̪, Ÿ̫, Ÿ̬, Ÿ̭, Ÿ̮, Ÿ̯, Ÿ̰, Ÿ̱, Ÿ̲, Ÿ̳, Ÿ̴, Ÿ̵, Ÿ̶, Ÿ̷, Ÿ̸, Ÿ̹, Ÿ̺, Ÿ̻, Ÿ̼, Ÿ̽, Ÿ̾, Ÿ̿}, % Cyr
8064 Y, Ÿ}, % Cyr
8065 Z = {Z̄, Ẑ, Z̃, Z̄, Z̅, Z̆, Ż, Z̈, Z̉, Z̊, Z̋, Ž, Z̍, Z̎, Z̏, Z̐, Z̑, Z̒, Z̓, Z̔, Z̕, Z̖, Z̗, Z̘, Z̙, Z̚, Z̛, Z̜, Z̝, Z̞, Z̟, Z̠, Z̡, Z̢, Ẓ, Z̤, Z̥, Z̦, Z̧, Z̨, Z̩, Z̪, Z̫, Z̬, Z̭, Z̮, Z̯, Z̰, Ẕ, Z̲, Z̳, Z̴, Z̵, Z̶, Z̷, Z̸, Z̹, Z̺, Z̻, Z̼, Z̽, Z̾, Z̿}, % a'
8066 a = {ā, ā̄, ā̂, ā̃, ā̄, ā̅, ā̆, ā̇, ā̈, ā̉, ā̊, ā̋, ā̌, ā̍, ā̎, ā̏, ā̐, ā̑, ā̒, ā̓, ā̔, ā̕, ā̖, ā̗, ā̘, ā̙, ā̚, ā̛, ā̜, ā̝, ā̞, ā̟, ā̠, ā̡, ā̢, ạ̄, ā̤, ḁ̄, ā̦, ā̧, ą̄, ā̩, ā̪, ā̫, ā̬, ā̭, ā̮, ā̯, ā̰, ā̱, ā̲, ā̳, ā̴, ā̵, ā̶, ā̷, ā̸, ā̹, ā̺, ā̻, ā̼, ā̽, ā̾, ā̿}, % a'
8067 a, ā̄}, % Cyr
8068 æ = {ǣ, æ̂, æ̃, ǣ, æ̅, æ̆, æ̇, æ̈, æ̉, æ̊, æ̋, æ̌, æ̍, æ̎, æ̏, æ̐, æ̑, æ̒, æ̓, æ̔, æ̕, æ̖, æ̗, æ̘, æ̙, æ̚, æ̛, æ̜, æ̝, æ̞, æ̟, æ̠, æ̡, æ̢, æ̣, æ̤, æ̥, æ̦, æ̧, æ̨, æ̩, æ̪, æ̫, æ̬, æ̭, æ̮, æ̯, æ̰, æ̱, æ̲, æ̳, æ̴, æ̵, æ̶, æ̷, æ̸, æ̹, æ̺, æ̻, æ̼, æ̽, æ̾, æ̿}, % Cyr
8069 æ}, % Cyr
8070 b = {b̄, b̂, b̃, b̄, b̅, b̆, ḃ, b̈, b̉, b̊, b̋, b̌, b̍, b̎, b̏, b̐, b̑, b̒, b̓, b̔, b̕, b̖, b̗, b̘, b̙, b̚, b̛, b̜, b̝, b̞, b̟, b̠, b̡, b̢, ḅ, b̤, b̥, b̦, b̧, b̨, b̩, b̪, b̫, b̬, b̭, b̮, b̯, b̰, ḇ, b̲, b̳, b̴, b̵, b̶, b̷, b̸, b̹, b̺, b̻, b̼, b̽, b̾, b̿}, % Cyr
8071 c = {ç̄, ç̂, ç̃, ç̄, ç̅, ç̆, ç̇, ç̈, ç̉, ç̊, ç̋, ç̌, ç̍, ç̎, ç̏, ç̐, ç̑, ç̒, ç̓, ç̔, ç̕, ç̖, ç̗, ç̘, ç̙, ç̚, ç̛, ç̜, ç̝, ç̞, ç̟, ç̠, ç̡, ç̢, ç̣, ç̤, ç̥, ç̦, ç̧, ç̨, ç̩, ç̪, ç̫, ç̬, ç̭, ç̮, ç̯, ç̰, ç̱, ç̲, ç̳, ç̴, ç̵, ç̶, ç̷, ç̸, ç̹, ç̺, ç̻, ç̼, ç̽, ç̾, ç̿}, % Cyr
8072 c, ç}, % Cyr
8073 d = {d̄, d̂, d̃, d̄, d̅, d̆, ḋ, d̈, d̉, d̊, d̋, ď, d̍, d̎, d̏, d̐, d̑, d̒, d̓, d̔, d̕, d̖, d̗, d̘, d̙, d̚, d̛, d̜, d̝, d̞, d̟, d̠, d̡, d̢, ḍ, d̤, d̥, d̦, ḑ, d̨, d̩, d̪, d̫, d̬, ḓ, d̮, d̯, d̰, ḏ, d̲, d̳, d̴, d̵, d̶, d̷, d̸, d̹, d̺, d̻, d̼, d̽, d̾, d̿}, % Cyr
8074 e = {è, è̄, è̂, è̃, è̄, è̅, è̆, è̇, è̈, è̉, è̊, è̋, è̌, è̍, è̎, è̏, è̐, è̑, è̒, è̓, è̔, è̕, è̖, è̗, è̘, è̙, è̚, è̛, è̜, è̝, è̞, è̟, è̠, è̡, è̢, ẹ̀, è̤, è̥, è̦, ȩ̀, ę̀, è̩, è̪, è̫, è̬, ḙ̀, è̮, è̯, ḛ̀, è̱, è̲, è̳, è̴, è̵, è̶, è̷, è̸, è̹, è̺, è̻, è̼, è̽, è̾, è̿}, % Cyr
8075 e, è̄, è̂}, % Cyr
8076 f = {f̄, f̂, f̃, f̄, f̅, f̆, ḟ, f̈, f̉, f̊, f̋, f̌, f̍, f̎, f̏, f̐, f̑, f̒, f̓, f̔, f̕, f̖, f̗, f̘, f̙, f̚, f̛, f̜, f̝, f̞, f̟, f̠, f̡, f̢, f̣, f̤, f̥, f̦, f̧, f̨, f̩, f̪, f̫, f̬, f̭, f̮, f̯, f̰, f̱, f̲, f̳, f̴, f̵, f̶, f̷, f̸, f̹, f̺, f̻, f̼, f̽, f̾, f̿}, % /f f
8077 g = {ğ̄, ğ̂, ğ̃, ğ̄, ğ̅, ğ̆, ğ̇, ğ̈, ğ̉, ğ̊, ğ̋, ğ̌, ğ̍, ğ̎, ğ̏, ğ̐, ğ̑, ğ̒, ğ̓, ğ̔, ğ̕, ğ̖, ğ̗, ğ̘, ğ̙, ğ̚, ğ̛, ğ̜, ğ̝, ğ̞, ğ̟, ğ̠, ğ̡, ğ̢, ğ̣, ğ̤, ğ̥, ğ̦, ģ̆, ğ̨, ğ̩, ğ̪, ğ̫, ğ̬, ğ̭, ğ̮, ğ̯, ğ̰, ğ̱, ğ̲, ğ̳, ğ̴, ğ̵, ğ̶, ğ̷, ğ̸, ğ̹, ğ̺, ğ̻, ğ̼, ğ̽, ğ̾, ğ̿}, % Cyr
8078 h = {h̄, ĥ, h̃, h̄, h̅, h̆, ḣ, ḧ, h̉, h̊, h̋, ȟ, h̍, h̎, h̏, h̐, h̑, h̒, h̓, h̔, h̕, h̖, h̗, h̘, h̙, h̚, h̛, h̜, h̝, h̞, h̟, h̠, h̡, h̢, ḥ, h̤, h̥, h̦, ḩ, h̨, h̩, h̪, h̫, h̬, h̭, ḫ, h̯, h̰, ẖ, h̲, h̳, h̴, h̵, h̶, h̷, h̸, h̹, h̺, h̻, h̼, h̽, h̾, h̿}, % Cyr
8079 h, h̄}, % Cyr
8080 i = {ī, î, ĩ, ī, i̅, ĭ, i̇, ï, ỉ, i̊, i̋, ǐ, i̍, i̎, ȉ, i̐, ȋ, i̒, i̓, i̔, i̕, i̖, i̗, i̘, i̙, i̚, i̛, i̜, i̝, i̞, i̟, i̠, i̡, i̢, ị, i̤, i̥, i̦, i̧, į, i̩, i̪, i̫, i̬, i̭, i̮, i̯, ḭ, i̱, i̲, i̳, i̴, i̵, i̶, i̷, i̸, i̹, i̺, i̻, i̼, i̽, i̾, i̿}, % Cyr
8081 i, ī}, % Cyr
8082 j = {j̄, ĵ, j̃, j̄, j̅, j̆, j̇, j̈, j̉, j̊, j̋, ǰ, j̍, j̎, j̏, j̐, j̑, j̒, j̓, j̔, j̕, j̖, j̗, j̘, j̙, j̚, j̛, j̜, j̝, j̞, j̟, j̠, j̡, j̢, j̣, j̤, j̥, j̦, j̧, j̨, j̩, j̪, j̫, j̬, j̭, j̮, j̯, j̰, j̱, j̲, j̳, j̴, j̵, j̶, j̷, j̸, j̹, j̺, j̻, j̼, j̽, j̾, j̿}, % Cyr
8083 j}, % Cyr
8084 k = {k̄, k̂, k̃, k̄, k̅, k̆, k̇, k̈, k̉, k̊, k̋, ǩ, k̍, k̎, k̏, k̐, k̑, k̒, k̓, k̔, k̕, k̖, k̗, k̘, k̙, k̚, k̛, k̜, k̝, k̞, k̟, k̠, k̡, k̢, ḳ, k̤, k̥, k̦, ķ, k̨, k̩, k̪, k̫, k̬, k̭, k̮, k̯, k̰, ḵ, k̲, k̳, k̴, k̵, k̶, k̷, k̸, k̹, k̺, k̻, k̼, k̽, k̾, k̿}, % Cyr
8085 l = {l̄, l̂, l̃, l̄, l̅, l̆, l̇, l̈, l̉, l̊, l̋, ľ, l̍, l̎, l̏, l̐, l̑, l̒, l̓, l̔, l̕, l̖, l̗, l̘, l̙, l̚, l̛, l̜, l̝, l̞, l̟, l̠, l̡, l̢, ḷ, l̤, l̥, l̦, ļ, l̨, l̩, l̪, l̫, l̬, ḽ, l̮, l̯, l̰, ḻ, l̲, l̳, l̴, l̵, l̶, l̷, l̸, l̹, l̺, l̻, l̼, l̽, l̾, l̿}, % l, l̄
8086 m = {m̄, m̂, m̃, m̄, m̅, m̆, ṁ, m̈, m̉, m̊, m̋, m̌, m̍, m̎, m̏, m̐, m̑, m̒, m̓, m̔, m̕, m̖, m̗, m̘, m̙, m̚, m̛, m̜, m̝, m̞, m̟, m̠, m̡, m̢, ṃ, m̤, m̥, m̦, m̧, m̨, m̩, m̪, m̫, m̬, m̭, m̮, m̯, m̰, m̱, m̲, m̳, m̴, m̵, m̶, m̷, m̸, m̹, m̺, m̻, m̼, m̽, m̾, m̿}, % n
8087 n = {ñ̄, ñ̂, ñ̃, ñ̄, ñ̅, ñ̆, ñ̇, ñ̈, ñ̉, ñ̊, ñ̋, ñ̌, ñ̍, ñ̎, ñ̏, ñ̐, ñ̑, ñ̒, ñ̓, ñ̔, ñ̕, ñ̖, ñ̗, ñ̘, ñ̙, ñ̚, ñ̛, ñ̜, ñ̝, ñ̞, ñ̟, ñ̠, ñ̡, ñ̢, ṇ̃, ñ̤, ñ̥, ñ̦, ņ̃, ñ̨, ñ̩, ñ̪, ñ̫, ñ̬, ṋ̃, ñ̮, ñ̯, ñ̰, ṉ̃, ñ̲, ñ̳, ñ̴, ñ̵, ñ̶, ñ̷, ñ̸, ñ̹, ñ̺, ñ̻, ñ̼, ñ̽, ñ̾, ñ̿}, % n
8088 o = {ò̄, ò̂, ò̃, ò̄, ò̅, ò̆, ò̇, ò̈, ò̉, ò̊, ò̋, ò̌, ò̍, ò̎, ò̏, ò̐, ò̑, ò̒, ò̓, ò̔, ò̕, ò̖, ò̗, ò̘, ò̙, ò̚, ờ, ò̜, ò̝, ò̞, ò̟, ò̠, ò̡, ò̢, ọ̀, ò̤, ò̥, ò̦, ò̧, ǫ̀, ò̩, ò̪, ò̫, ò̬, ò̭, ò̮, ò̯, ò̰, ò̱, ò̲, ò̳, ò̴, ò̵, ò̶, ò̷, ò̸, ò̹, ò̺, ò̻, ò̼, ò̽, ò̾, ò̿}, % Cyr
8089 o, ò̄, ò̂, ò̃}, % Cyr
8090 p = {p̄, p̂, p̃, p̄, p̅, p̆, ṗ, p̈, p̉, p̊, p̋, p̌, p̍, p̎, p̏, p̐, p̑, p̒, p̓, p̔, p̕, p̖, p̗, p̘, p̙, p̚, p̛, p̜, p̝, p̞, p̟, p̠, p̡, p̢, p̣, p̤, p̥, p̦, p̧, p̨, p̩, p̪, p̫, p̬, p̭, p̮, p̯, p̰, p̱, p̲, p̳, p̴, p̵, p̶, p̷, p̸, p̹, p̺, p̻, p̼, p̽, p̾, p̿}, % Cyr
8091 p, p̄}, % Cyr
8092 q = {q̄, q̂, q̃, q̄, q̅, q̆, q̇, q̈, q̉, q̊, q̋, q̌, q̍, q̎, q̏, q̐, q̑, q̒, q̓, q̔, q̕, q̖, q̗, q̘, q̙, q̚, q̛, q̜, q̝, q̞, q̟, q̠, q̡, q̢, q̣, q̤, q̥, q̦, q̧, q̨, q̩, q̪, q̫, q̬, q̭, q̮, q̯, q̰, q̱, q̲, q̳, q̴, q̵, q̶, q̷, q̸, q̹, q̺, q̻, q̼, q̽, q̾, q̿}, % Cyr
8093 r = {r̄, r̂, r̃, r̄, r̅, r̆, ṙ, r̈, r̉, r̊, r̋, ř, r̍, r̎, ȑ, r̐, ȓ, r̒, r̓, r̔, r̕, r̖, r̗, r̘, r̙, r̚, r̛, r̜, r̝, r̞, r̟, r̠, r̡, r̢, ṛ, r̤, r̥, r̦, ŗ, r̨, r̩, r̪, r̫, r̬, r̭, r̮, r̯, r̰, ṟ, r̲, r̳, r̴, r̵, r̶, r̷, r̸, r̹, r̺, r̻, r̼, r̽, r̾, r̿}, % Cyr
8094 s = {s̄, ŝ, s̃, s̄, s̅, s̆, ṡ, s̈, s̉, s̊, s̋, š, s̍, s̎, s̏, s̐, s̑, s̒, s̓, s̔, s̕, s̖, s̗, s̘, s̙, s̚, s̛, s̜, s̝, s̞, s̟, s̠, s̡, s̢, ṣ, s̤, s̥, ș, ş, s̨, s̩, s̪, s̫, s̬, s̭, s̮, s̯, s̰, s̱, s̲, s̳, s̴, s̵, s̶, s̷, s̸, s̹, s̺, s̻, s̼, s̽, s̾, s̿}, % Cyr
8095 s}, % Cyr
8096 t = {t̄, t̂, t̃, t̄, t̅, t̆, ṫ, ẗ, t̉, t̊, t̋, ť, t̍, t̎, t̏, t̐, t̑, t̒, t̓, t̔, t̕, t̖, t̗, t̘, t̙, t̚, t̛, t̜, t̝, t̞, t̟, t̠, t̡, t̢, ṭ, t̤, t̥, ț, ţ, t̨, t̩, t̪, t̫, t̬, ṱ, t̮, t̯, t̰, ṯ, t̲, t̳, t̴, t̵, t̶, t̷, t̸, t̹, t̺, t̻, t̼, t̽, t̾, t̿}, % t
8097 u = {ù, ú, û, ü, ū, ū̄, ū̂, ū̃, ū̄, ū̅, ū̆, ū̇, ṻ, ū̉, ū̊, ū̋, ū̌, ū̍, ū̎, ū̏, ū̐, ū̑, ū̒, ū̓, ū̔, ū̕, ū̖, ū̗, ū̘, ū̙, ū̚, ư̄, ū̜, ū̝, ū̞, ū̟, ū̠, ū̡, ū̢, ụ̄, ṳ̄, ū̥, ū̦, ū̧, ų̄, ū̩, ū̪, ū̫, ū̬, ṷ̄, ū̮, ū̯, ṵ̄, ū̱, ū̲, ū̳, ū̴, ū̵, ū̶, ū̷, ū̸, ū̹, ū̺, ū̻, ū̼, ū̽, ū̾, ū̿}, % Cyr
8098 v = {v̄, v̂, ṽ, v̄, v̅, v̆, v̇, v̈, v̉, v̊, v̋, v̌, v̍, v̎, v̏, v̐, v̑, v̒, v̓, v̔, v̕, v̖, v̗, v̘, v̙, v̚, v̛, v̜, v̝, v̞, v̟, v̠, v̡, v̢, ṿ, v̤, v̥, v̦, v̧, v̨, v̩, v̪, v̫, v̬, v̭, v̮, v̯, v̰, v̱, v̲, v̳, v̴, v̵, v̶, v̷, v̸, v̹, v̺, v̻, v̼, v̽, v̾, v̿}, % Cyr
8099 w = {w̄, ŵ, w̃, w̄, w̅, w̆, ẇ, ẅ, w̉, ẘ, w̋, w̌, w̍, w̎, w̏, w̐, w̑, w̒, w̓, w̔, w̕, w̖, w̗, w̘, w̙, w̚, w̛, w̜, w̝, w̞, w̟, w̠, w̡, w̢, ẉ, w̤, w̥, w̦, w̧, w̨, w̩, w̪, w̫, w̬, w̭, w̮, w̯, w̰, w̱, w̲, w̳, w̴, w̵, w̶, w̷, w̸, w̹, w̺, w̻, w̼, w̽, w̾, w̿}, % Cyr
8100 w}, % Cyr
8101 x = {x̄, x̂, x̃, x̄, x̅, x̆, ẋ, ẍ, x̉, x̊, x̋, x̌, x̍, x̎, x̏, x̐, x̑, x̒, x̓, x̔, x̕, x̖, x̗, x̘, x̙, x̚, x̛, x̜, x̝, x̞, x̟, x̠, x̡, x̢, x̣, x̤, x̥, x̦, x̧, x̨, x̩, x̪, x̫, x̬, x̭, x̮, x̯, x̰, x̱, x̲, x̳, x̴, x̵, x̶, x̷, x̸, x̹, x̺, x̻, x̼, x̽, x̾, x̿}, % Cyr
8102 x, x̄}, % Cyr
8103 y = {ý, ŷ, ŷ̄, ŷ̂, ŷ̃, ŷ̄, ŷ̅, ŷ̆, ŷ̇, ŷ̈, ŷ̉, ŷ̊, ŷ̋, ŷ̌, ŷ̍, ŷ̎, ŷ̏, ŷ̐, ŷ̑, ŷ̒, ŷ̓, ŷ̔, ŷ̕, ŷ̖, ŷ̗, ŷ̘, ŷ̙, ŷ̚, ŷ̛, ŷ̜, ŷ̝, ŷ̞, ŷ̟, ŷ̠, ŷ̡, ŷ̢, ỵ̂, ŷ̤, ŷ̥, ŷ̦, ŷ̧, ŷ̨, ŷ̩, ŷ̪, ŷ̫, ŷ̬, ŷ̭, ŷ̮, ŷ̯, ŷ̰, ŷ̱, ŷ̲, ŷ̳, ŷ̴, ŷ̵, ŷ̶, ŷ̷, ŷ̸, ŷ̹, ŷ̺, ŷ̻, ŷ̼, ŷ̽, ŷ̾, ŷ̿}, % Cyr
8104 y, ŷ, ŷ̄, ŷ̂, ŷ̃}, % Cyr
8105 z = {Z̄, Ẑ, Z̃, Z̄, Z̅, Z̆, Ż, Z̈, Z̉, Z̊, Z̋, Ž, Z̍, Z̎, Z̏, Z̐, Z̑, Z̒, Z̓, Z̔, Z̕, Z̖, Z̗, Z̘, Z̙, Z̚, Z̛, Z̜, Z̝, Z̞, Z̟, Z̠, Z̡, Z̢, Ẓ, Z̤, Z̥, Z̦, Z̧, Z̨, Z̩, Z̪, Z̫, Z̬, Z̭, Z̮, Z̯, Z̰, Ẕ, Z̲, Z̳, Z̴, Z̵, Z̶, Z̷, Z̸, Z̹, Z̺, Z̻, Z̼, Z̽, Z̾, Z̿}, % Cyr
8106 Z̄, Ẑ, Z̃}, % Cyr
8107 Γ = {Γ̄, Γ̂, Γ̃, Γ̄, Γ̅, Γ̆, Γ̇, Γ̈, Γ̉, Γ̊, Γ̋, Γ̌, Γ̍, Γ̎, Γ̏, Γ̐, Γ̑, Γ̒, Γ̓, Γ̔, Γ̕, Γ̖, Γ̗, Γ̘, Γ̙, Γ̚, Γ̛, Γ̜, Γ̝, Γ̞, Γ̟, Γ̠, Γ̡, Γ̢, Γ̣, Γ̤, Γ̥, Γ̦, Γ̧, Γ̨, Γ̩, Γ̪, Γ̫, Γ̬, Γ̭, Γ̮, Γ̯, Γ̰, Γ̱, Γ̲, Γ̳, Γ̴, Γ̵, Γ̶, Γ̷, Γ̸, Γ̹, Γ̺, Γ̻, Γ̼, Γ̽, Γ̾, Γ̿}, % Cyr
8108 Ж = {Ж̄, Ж̂, Ж̃, Ж̄, Ж̅, Ӂ, Ж̇, Ӝ, Ж̉, Ж̊, Ж̋, Ж̌, Ж̍, Ж̎, Ж̏, Ж̐, Ж̑, Ж̒, Ж̓, Ж̔, Ж̕, Ж̖, Ж̗, Ж̘, Ж̙, Ж̚, Ж̛, Ж̜, Ж̝, Ж̞, Ж̟, Ж̠, Ж̡, Ж̢, Ж̣, Ж̤, Ж̥, Ж̦, Ж̧, Ж̨, Ж̩, Ж̪, Ж̫, Ж̬, Ж̭, Ж̮, Ж̯, Ж̰, Ж̱, Ж̲, Ж̳, Ж̴, Ж̵, Ж̶, Ж̷, Ж̸, Ж̹, Ж̺, Ж̻, Ж̼, Ж̽, Ж̾, Ж̿}, % Cyr
8109 З = {З̄, З̂, З̃, З̄, З̅, З̆, З̇, Ӟ, З̉, З̊, З̋, З̌, З̍, З̎, З̏, З̐, З̑, З̒, З̓, З̔, З̕, З̖, З̗, З̘, З̙, З̚, З̛, З̜, З̝, З̞, З̟, З̠, З̡, З̢, З̣, З̤, З̥, З̦, З̧, З̨, З̩, З̪, З̫, З̬, З̭, З̮, З̯, З̰, З̱, З̲, З̳, З̴, З̵, З̶, З̷, З̸, З̹, З̺, З̻, З̼, З̽, З̾, З̿}, % Cyr
8110 Л = {Л̄, Л̂, Л̃, Л̄, Л̅, Л̆, Л̇, Л̈, Л̉, Л̊, Л̋, Л̌, Л̍, Л̎, Л̏, Л̐, Л̑, Л̒, Л̓, Л̔, Л̕, Л̖, Л̗, Л̘, Л̙, Л̚, Л̛, Л̜, Л̝, Л̞, Л̟, Л̠, Л̡, Л̢, Л̣, Л̤, Л̥, Л̦, Л̧, Л̨, Л̩, Л̪, Л̫, Л̬, Л̭, Л̮, Л̯, Л̰, Л̱, Л̲, Л̳, Л̴, Л̵, Л̶, Л̷, Л̸, Л̹, Л̺, Л̻, Л̼, Л̽, Л̾, Л̿}, % Cyr
8111 П = {П̄, П̂, П̃, П̄, П̅, П̆, П̇, П̈, П̉, П̊, П̋, П̌, П̍, П̎, П̏, П̐, П̑, П̒, П̓, П̔, П̕, П̖, П̗, П̘, П̙, П̚, П̛, П̜, П̝, П̞, П̟, П̠, П̡, П̢, П̣, П̤, П̥, П̦, П̧, П̨, П̩, П̪, П̫, П̬, П̭, П̮, П̯, П̰, П̱, П̲, П̳, П̴, П̵, П̶, П̷, П̸, П̹, П̺, П̻, П̼, П̽, П̾, П̿}, % Cyr
8112 Y = {Ÿ̄, Ÿ̂, Ÿ̃, Ÿ̄, Ÿ̅, Ÿ̆, Ÿ̇, Ÿ̈, Ÿ̉, Ÿ̊, Ÿ̋, Ÿ̌, Ÿ̍, Ÿ̎, Ÿ̏, Ÿ̐, Ÿ̑, Ÿ̒, Ÿ̓, Ÿ̔, Ÿ̕, Ÿ̖, Ÿ̗, Ÿ̘, Ÿ̙, Ÿ̚, Ÿ̛, Ÿ̜, Ÿ̝, Ÿ̞, Ÿ̟, Ÿ̠, Ÿ̡, Ÿ̢, Ỵ̈, Ÿ̤, Ÿ̥, Ÿ̦, Ÿ̧, Ÿ̨, Ÿ̩, Ÿ̪, Ÿ̫, Ÿ̬, Ÿ̭, Ÿ̮, Ÿ̯, Ÿ̰, Ÿ̱, Ÿ̲, Ÿ̳, Ÿ̴, Ÿ̵, Ÿ̶, Ÿ̷, Ÿ̸, Ÿ̹, Ÿ̺, Ÿ̻, Ÿ̼, Ÿ̽, Ÿ̾, Ÿ̿}, % Cyr
8113 Ч = {Ч̄, Ч̂, Ч̃, Ч̄, Ч̅, Ч̆, Ч̇, Ӵ, Ч̉, Ч̊, Ч̋, Ч̌, Ч̍, Ч̎, Ч̏, Ч̐, Ч̑, Ч̒, Ч̓, Ч̔, Ч̕, Ч̖, Ч̗, Ч̘, Ч̙, Ч̚, Ч̛, Ч̜, Ч̝, Ч̞, Ч̟, Ч̠, Ч̡, Ч̢, Ч̣, Ч̤, Ч̥, Ч̦, Ч̧, Ч̨, Ч̩, Ч̪, Ч̫, Ч̬, Ч̭, Ч̮, Ч̯, Ч̰, Ч̱, Ч̲, Ч̳, Ч̴, Ч̵, Ч̶, Ч̷, Ч̸, Ч̹, Ч̺, Ч̻, Ч̼, Ч̽, Ч̾, Ч̿}, % Cyr
8114 Ъ = {Ъ̄, Ъ̂, Ъ̃, Ъ̄, Ъ̅, Ъ̆, Ъ̇, Ъ̈, Ъ̉, Ъ̊, Ъ̋, Ъ̌, Ъ̍, Ъ̎, Ъ̏, Ъ̐, Ъ̑, Ъ̒, Ъ̓, Ъ̔, Ъ̕, Ъ̖, Ъ̗, Ъ̘, Ъ̙, Ъ̚, Ъ̛, Ъ̜, Ъ̝, Ъ̞, Ъ̟, Ъ̠, Ъ̡, Ъ̢, Ъ̣, Ъ̤, Ъ̥, Ъ̦, Ъ̧, Ъ̨, Ъ̩, Ъ̪, Ъ̫, Ъ̬, Ъ̭, Ъ̮, Ъ̯, Ъ̰, Ъ̱, Ъ̲, Ъ̳, Ъ̴, Ъ̵, Ъ̶, Ъ̷, Ъ̸, Ъ̹, Ъ̺, Ъ̻, Ъ̼, Ъ̽, Ъ̾, Ъ̿}, % Cyr
8115 Ъ = {Ъ̄, Ъ̂, Ъ̃, Ъ̄, Ъ̅, Ъ̆, Ъ̇, Ъ̈, Ъ̉, Ъ̊, Ъ̋, Ъ̌, Ъ̍, Ъ̎, Ъ̏, Ъ̐, Ъ̑, Ъ̒, Ъ̓, Ъ̔, Ъ̕, Ъ̖, Ъ̗, Ъ̘, Ъ̙, Ъ̚, Ъ̛, Ъ̜, Ъ̝, Ъ̞, Ъ̟, Ъ


```

8236 {,}= { ,500},
8237 := { ,500},
8238 ;= { ,500},
8239 != { ,100},
8240 ? = { ,200},
8241 @ = {50,50},
8242 ~ = {200,250},
8243 \% = {50,50},
8244 * = {300,300},
8245 + = {250,250},
8246 - = {400,500}, % /hyphen
8247 – = {400,300}, % /endash
8248 — = {300,200}, % /emdash
8249 _ = {200,200}, % /underscore
8250 / = {200,300},
8251 /backslash = {200,300},
8252 ' = {300,400}, % /quotesingle
8253 ‘ = {500,700}, ’ = {500,600},
8254 “ = {500,300}, ” = {200,600},
8255 , = {400,400}, ,, = {400,400},
8256 ‹ = {400,400}, › = {300,500},
8257 « = {300,200}, » = {100,400},
8258 ¡ = {100, }, ¡ = {100, },
8259 ( = {300, }, ) = { ,300},
8260 < = {200,100}, > = {100,200},
8261 /braceleft = {400,200}, /braceright = {200,400},
8262 /angleleft = {400, }, /angleright = { ,400},
8263 † = {100,100},
8264 ‡ = { 80, 80},
8265 • = {200,200},
8266 · = {400,450}, % / periodcentered
8267 °C = { 80, 50},
8268 ℄ = { , 50},
8269 ° = {400,400},
8270 ™ = {100,200},
8271 © = {100,100},
8272 ® = {100,100},
8273 ª = {100,200},
8274 º = {100,200},
8275 ¹ = {200,250},
8276 º = { 50,100},
8277 ¸ = { 50,100},
8278 ¬ = {200, },
8279 − = {300,300},
8280 ± = {150,200},
8281 × = {150,250},
8282 ÷ = {150,250},
8283 € = {100, },
8284 /one.oldstyle = {100,100},
8285 /two.oldstyle = { 50, 50},
8286 /three.oldstyle = { 30, 80},
8287 /four.oldstyle = { 50, 50},
8288 /seven.oldstyle = { 50, 80},
8289 Γ = { ,180}, % /Gamma
8290 Δ = {100,100}, % /Delta
8291 Θ = { 50, 50}, % /Theta
8292 Λ = {100,100}, % /Lambda
8293 % Ξ = {,}, % /Xi
8294 % Π = {,}, % /Pi
8295 Σ = { 50, 50}, % /Sigma
8296 Υ = {100,100}, % /Upsilon
8297 Φ = { 50, 50}, % /Phi
8298 Ψ = { 50, 50}, % /Psi
8299 % Ω = {,}, % /Omega
8300 }

```

```
8301
8302 \SetProtrusion
8303   [ name      = LMR-it ]
8304   { encoding = {EU1,EU2,TU},
8305     family   = Latin Modern Roman,
8306     shape    = {it,s1}      }
8307   {
8308     A = {125,100},
8309     Æ = {125,-55},
8310     B = {90,-40},
8311     C = {145,-75},
8312     D = {75, -28},
8313     E = {80,-55},
8314     F = {85,-80},
8315     G = {153,-15},
8316     H = {73,-60},
8317     I = {140,-120},
8318     IJ = {140,-80},
8319     J = {135,-80},
8320     K = {70,-30},
8321     L = {87, 40},
8322     M = {67,-45},
8323     N = {75,-55},
8324     O = {150,-30},
8325     Œ = {150,-55},
8326     P = {82,-50},
8327     Q = {150,-30},
8328     R = {75, 15},
8329     S = {90,-65},
8330     $ = {100,-20},
8331     T = {220,-85},
8332     U = {230,-55},
8333     V = {260,-60},
8334     W = {185,-55},
8335     X = {70,-30},
8336     Y = {250,-60},
8337     Z = {90,-60},
8338     a = {150,-10},
8339     b = {170,  },
8340     c = {173,-10},
8341     d = {150,-55},
8342     e = {180, },
8343     f = {  ,-250},
8344     g = {150,-10},
8345     h = {100, },
8346     i = {210, },
8347     ij = {210,-40},
8348     j = {  ,-40},
8349     k = {110,-50},
8350     l = {240,-110},
8351     m = {80, },
8352     n = {115, },
8353     o = {155, },
8354     q = {170,-40},
8355     r = {155,-40},
8356     s = {130, },
8357     t = {230,-10},
8358     u = {120, },
8359     v = {140,-25},
8360     w = {98,-20},
8361     x = {65,-40},
8362     y = {130,-20},
8363     z = {110,-80},
8364     0 = {170,-85},
8365     1 = {230,110},
```

8366 2 = {130,-70},
8367 3 = {140,-70},
8368 4 = {130,80},
8369 5 = {160, },
8370 6 = {175,-30},
8371 7 = {250,-150},
8372 8 = {130,-40},
8373 9 = {155,-80},
8374 . = { ,500},
8375 {,}= { ,450},
8376 := { ,300},
8377 ; = { ,300},
8378 & = {130,30},
8379 \% = {180,50},
8380 * = {380,20},
8381 + = {180,200},
8382 @ = {180,10},
8383 ~ = {200,150},
8384 (= {300, },) = { ,70},
8385 / = {100,100},
8386 - = {500,300}, % /hyphen
8387 – = {500,300}, % /endash
8388 — = {400,170}, % /emdash
8389 _ = {100,200}, % /underscore
8390 ' = {300,400}, % /quotesingle
8391 " = {500,300},
8392 ‘ = {800,200}, ’ = {800,-20},
8393 “ = {540,100}, ” = {500,100},
8394 , = {300,700}, ,, = {200,600},
8395 ‹ = {500,300}, › = {400,400},
8396 « = {400,100}, » = {200,300},
8397 ì = {200, }, î = {200, },
8398 < = {300,100}, > = {200,100},
8399 /backslash = {300,300},
8400 /braceleft = {400,100}, /braceright = {200,200},
8401 † = {200, 80},
8402 ‡ = {120, 80},
8403 • = {220,100},
8404 · = {550,300}, % / periodcentered
8405 °C = {170, },
8406 © = {100, 50},
8407 ¶ = {200, },
8408 ° = {500,300},
8409 ™ = {200, 70},
8410 © = { 50, 70},
8411 ® = { 50, 70},
8412 ª = {140,100},
8413 º = {140,100},
8414 ¹ = {400,150},
8415 ² = {250, 80},
8416 ³ = {250, 80},
8417 ¬ = {250, 80},
8418 − = {300,200},
8419 ± = {150,170},
8420 × = {200,200},
8421 ÷ = {200,200},
8422 € = {150, },
8423 /one.oldstyle = {100,100},
8424 /two.oldstyle = {100, 80},
8425 /three.oldstyle = { 80, 50},
8426 /four.oldstyle = { 80, 80},
8427 /five.oldstyle = { 50, },
8428 /six.oldstyle = { 50, },
8429 /seven.oldstyle = { 80, 80},
8430 /eight.oldstyle = { 50, },

```

8431   Γ = {100,120}, % /Gamma
8432   Δ = {120,100}, % /Delta
8433   Θ = {120, 50}, % /Theta
8434   Λ = {130,100}, % /Lambda
8435   Ξ = {100,},    % /Xi
8436   Π = {100,},    % /Pi
8437   Σ = {100, 50}, % /Sigma
8438   Υ = {180,100}, % /Upsilon
8439   Φ = {130, 70}, % /Phi
8440   Ψ = {130, 50}, % /Psi
8441   Ω = { 50,},    % /Omega
8442   }
8443   /LatinModernRoman
8444   (*CharisSIL)
8445   \SetProtrusion
8446   [ name      = Charis-default ]
8447   { encoding = {EU1,EU2,TU},
8448     family    = Charis SIL }
8449   {
8450     A = {50,50},
8451     Æ = {50,50},
8452     C = {50, },
8453     D = { ,50},
8454     F = { ,50},
8455     G = {50, },
8456     J = {100, },
8457     K = { ,50},
8458     L = { ,50},
8459     Ḷ = { ,100},
8460     O = {50,50},
8461     Œ = {50, },
8462     P = { ,50},
8463     Q = {50,70},
8464     R = { ,50},
8465     Œ = { ,40}, % capital sharp s
8466     T = {50,50},
8467     V = {50,50},
8468     W = {50,50},
8469     X = {50,50},
8470     Y = {50,50},
8471     k = { ,50},
8472     ḷ = { ,150},
8473     r = { ,50},
8474     t = { ,50},
8475     v = {50,50},
8476     w = {50,50},
8477     x = {50,50},
8478     y = { ,50},
8479     1 = {150,150},
8480     2 = {50,50},
8481     3 = {50, },
8482     4 = {100,50},
8483     6 = {50, },
8484     7 = {50,80},
8485     9 = {50,50},
8486     . = { ,600},
8487     {,} = { ,500},
8488     := { ,400},
8489     ; = { ,300},
8490     ! = { ,100},
8491     ? = { ,200},
8492     @ = {50,50},
8493     ~ = {200,250},
8494     \% = { ,50},
8495     * = {300,300},

```

8496 + = {200,250},
 8497 / = { ,200},
 8498 /backslash = {150,200},
 8499 | = {200,200},
 8500 - = {400,500}, % hyphen
 8501 – = {200,300}, % endash
 8502 — = {150,250}, % emdash
 8503 ⎯ = {200,200}, % Horizontal Bar = \texttwelveudash
 8504 - = {150,150}, % Figure Dash = \textthreequartersemdash
 8505 _ = {100,100},
 8506 {=} = {100,100},
 8507 ‘ = {300,400}, ’ = {300,400},
 8508 “ = {300,300}, ” = {300,300},
 8509 , = {400,400}, „ = {300,300},
 8510 ‹ = {400,300}, › = {300,400},
 8511 ‹‹ = {200,200}, ›› = {150,300},
 8512 ¡ = {100, }, ¿ = {100, },
 8513 (= {200, },) = { ,200},
 8514 < = {200,150}, > = {100,200},
 8515 [= {100, },] = { ,100},
 8516 /braceleft = {200, }, /braceright = { ,300},
 8517 † = { 80, 80},
 8518 ‡ = {100,100},
 8519 • = {200,200},
 8520 ° = {150,200},
 8521 ™ = {150,150},
 8522 ¢ = { 50, },
 8523 £ = { 50, },
 8524 † = {200,200},
 8525 © = {100,100},
 8526 ® = {100,100},
 8527 º = {100,200},
 8528 ° = {200,200},
 8529 ¬ = {200, 50},
 8530 μ = { ,100},
 8531 ¶ = { ,100},
 8532 · = {300,400},
 8533 ¹ = {200,300},
 8534 ² = {100,200},
 8535 ³ = {100,200},
 8536 € = {100, },
 8537 ± = {150,200},
 8538 × = {200,200},
 8539 ÷ = {250,250},
 8540 /minus = {200,200},
 8541 − = {200,200},
 8542 % Cyrillic
 8543 Б = { ,50},
 8544 Г = { ,130},
 8545 Ж = {50,50},
 8546 З = {30,50},
 8547 Л = {50, },
 8548 У = {50,50},
 8549 Ф = {50,50},
 8550 Ч = {100, },
 8551 Ъ = { ,50},
 8552 Ь = { ,50},
 8553 Э = {50,50},
 8554 Ю = { ,40},
 8555 Я = {50, },
 8556 В = {50,50},
 8557 € = {50, },
 8558 Ъ = {50,100},
 8559 € = {50, },
 8560 Ъ = {50,50},


```

8561  Ѓ = { ,50},
8562  Ɔ = {50,50},
8563  Ɔ = {100,100},
8564  Ɔ = {50,50},
8565  Ѓ = { ,50},
8566  Ѓ = { ,50},
8567  Ѓ = {50,80},
8568  Ɔ = { ,80},
8569  Ɔ = {50,50},
8570  Ɔ = {50, },
8571  Ɔ = {50,40},
8572  Ɔ = { ,50},
8573  Ɔ = {50, },
8574  Ɔ = { ,50},
8575  Ɔ = { ,50},
8576  Ɔ = { ,100},
8577  Ɔ = {50,50},
8578  Ɔ = { ,70},
8579  Ɔ = { ,50},
8580  Ɔ = {50, },
8581  Ɔ = {50,50},
8582  Ɔ = {50,50},
8583  Ɔ = {50, },
8584  Ɔ = { ,50},
8585  Ɔ = { ,50},
8586  Ɔ = { ,50},
8587  Ɔ = {50, },
8588  Ɔ = {50, },
8589  Ɔ = { ,50},
8590  Ɔ = { ,50},
8591  Ɔ = {50,50},
8592  Ɔ = {50, },
8593  Ɔ = { ,50},
8594  Ɔ = {50,50},
8595  Ɔ = { ,50},
8596  Ɔ = { ,50},
8597  Ɔ = { ,100},
8598  Ɔ = {100,100},
8599  Ɔ = {50,50},
8600  Ɔ = {50,70},
8601  Ɔ = { ,70},
8602  Ɔ = {50,30},
8603  Ɔ = { ,50},
8604  Ɔ = { ,50},
8605  % Д П Ц Ш Щ Ъ Ы Ь Ѓ Ѡ ѡ ѣ ѥ Ѧ ѧ Ѩ ѩ Ѭ ѭ
8606  % в д ж з и м н п ц ш щ ю ѣ ѥ Ѧ ѧ Ѩ ѩ Ѭ ѭ Ѯ ѯ Ѱ ѱ
8607  % Greek
8608  Δ = {50,50},
8609  Ψ = {50,50},
8610  γ = {70,70},
8611  λ = {40,70},
8612  π = {40,50},
8613  ρ = { ,50},
8614  σ = { ,50},
8615  χ = {50,50},

8616 }
8617
8618 \SetProtrusion
8619 [ name = Charis-it ]
8620 { encoding = {EU1,EU2,TU},
8621 family = Charis SIL,
8622 shape = {it,sl} }
8623 {
8624 C = {50, },
8625 G = {50, },

```

8626 J = {50, },
8627 L = {50,50},
8628 O = {50, },
8629 Œ = {50, },
8630 Q = {50, },
8631 S = {50, },
8632 \$ = {50, },
8633 T = {70, },
8634 o = {50,50},
8635 p = { ,50},
8636 q = {50, },
8637 t = { ,50},
8638 w = { ,50},
8639 y = { ,50},
8640 1 = {150,100},
8641 3 = {50, },
8642 4 = {100, },
8643 6 = {50, },
8644 7 = {100, },
8645 . = { ,700},
8646 {,} = { ,600},
8647 : = { ,400},
8648 ; = { ,400},
8649 ? = { ,150},
8650 & = { ,80},
8651 \% = {50,50},
8652 * = {300,200},
8653 + = {250,250},
8654 @ = {80,50},
8655 ~ = {150,150},
8656 / = { ,150},
8657 /backslash = {150,150},
8658 - = {300,400}, % hyphen
8659 - = {200,300}, % endash
8660 — = {150,200}, % emdash
8661 _ = { ,100},
8662 {=} = {200,200},
8663 ± = {150,200},
8664 × = {250,250},
8665 ÷ = {250,250},
8666 ° = {150,200},
8667 · = {300,400},
8668 ‘ = {400,200}, ’ = {400,200},
8669 “ = {300,200}, ” = {400,200},
8670 , = {200,500}, „ = {150,500},
8671 ‹ = {300,400}, › = {200,500},
8672 « = {200,300}, » = {150,400},
8673 (= {200, },) = { ,200},
8674 < = {200,200}, > = {200,200},
8675 /braceleft = {300, }, /braceright = { ,200},
8676 % Cyrillic
8677 Ж = {50,30},
8678 Л = {50, },
8679 У = {50,30},
8680 Ф = {50, },
8681 Ч = {100, },
8682 Ъ = { ,50},
8683 Ь = { ,50},
8684 Э = {50,50},
8685 Я = {50, },
8686 В = {50,50},
8687 Ъ = {50,50},
8688 Ъ = {140,100},
8689 Ъ = {70,50},
8690 Ъ = {50,80},

```

8691   Ḥ = { ,80},
8692   Ŧ = {50,50},
8693   Γ = {50,50},
8694   Δ = {50,30},
8695   Μ = {50, },
8696   Φ = {50, },
8697   Ч = {50, },
8698   Ъ = { ,50},
8699   Ь = { ,50},
8700   Э = { ,50},
8701   Я = {50, },
8702   Љ = {50,50},
8703   Њ = { ,50},
8704   В = {50,50},
8705   Ъ = { ,50},
8706   Ъ = {140,100},
8707   ʒ = {70,50},
8708   ʟ = {50,70},
8709   Ɔ = { ,70},
8710   % Greek
8711   Γ = { ,130},
8712   Δ = {50,50},
8713   Ψ = {50,50},
8714   γ = {70,70},
8715   λ = {40,70},
8716   π = {40,50},
8717   ρ = { ,50},
8718   σ = { ,50},
8719   χ = {50,50},
8720   }

```

The small caps glyph names in Charis SIL have changed with version 5.0 of the font. We try to get the names right both with LuaTeX (where we can simply query the font version) and with XeTeX (where we check for glyph name).

```

8721
8722 % quick and dirty -- maybe we'll promote this to a
8723 % regular key some time
8724 \define@key{MT@pr@c}{command}{\csname #1\endcsname}
8725
8726 % glyph names have changed with version 5.0 of Charis SIL:
8727 % before: /a.SC, /b.SC, ...
8728 % after: /a.sc, /b.sc, ...
8729 \ifx\MT@lua\undefined
8730   \gdef\MT@get@CHARIS@SC{
8731     % test whether glyph "a.sc" exists
8732     \ifnum\numexpr\XeTeXglyphindex "a.sc"\relax > 0
8733       \gdef\MT@CHARIS@SC{sc}%
8734     \else
8735       \gdef\MT@CHARIS@SC{SC}%
8736     \fi
8737   }
8738 \else
8739   \gdef\MT@get@CHARIS@SC{
8740     \gdef\MT@CHARIS@SC{\MT@lua{
8741       % check font version
8742 % -- why doesn't this work?:
8743 %   f = font.getfont(font.current());
8744 %   i = fontloader.info(f.filename);
8745 %   if (tonumber(i.version) < 5) then;
8746 %     if (tonumber(fontloader.info(font.getfont(font.current()).filename).version) < 5) then;
8747 %       tex.print("SC");
8748 %     else;
8749 %       tex.print("sc");
8750 %     end

```

```

8751   }}
8752   }
8753 \fi
8754
8755 \SetProtrusion
8756   [ name      = Charis-sc,
8757     load      = Charis-default,
8758     command   = {MT@get@CHARIS@SC} ]
8759   { encoding = {EU1,EU2,TU},
8760     family   = Charis SIL,
8761     shape    = {sc} }
8762   {
8763     % A = {100,100}, % etc., doesn't work with \textsc
8764     /a.\MT@CHARIS@SC = {100,100},
8765     /c.\MT@CHARIS@SC = {50, },
8766     /d.\MT@CHARIS@SC = { ,50},
8767     /f.\MT@CHARIS@SC = { ,50},
8768     /g.\MT@CHARIS@SC = {50, },
8769     /j.\MT@CHARIS@SC = {100, },
8770     /k.\MT@CHARIS@SC = { ,50},
8771     /l.\MT@CHARIS@SC = { ,50},
8772     /f.l.\MT@CHARIS@SC = { ,50},
8773     /o.\MT@CHARIS@SC = {50,50},
8774     /oe.\MT@CHARIS@SC = {50, },
8775     /q.\MT@CHARIS@SC = {50,70},
8776     /r.\MT@CHARIS@SC = { ,50},
8777     /t.\MT@CHARIS@SC = {50,100},
8778     /v.\MT@CHARIS@SC = {50,50},
8779     /w.\MT@CHARIS@SC = {50,50},
8780     /x.\MT@CHARIS@SC = {50,50},
8781     /y.\MT@CHARIS@SC = {50,50}
8782   }
8783 </CharisSIL>
8784 < *PalatinoLinotype >
8785 \SetProtrusion
8786   [ name      = palatino-default ]
8787   { encoding = {EU1,EU2,TU},
8788     family   = {PalatinoLinotype} }
8789   {
8790     A = {50,50},
8791     D = { ,50},
8792     J = {50, },
8793     K = { ,50},
8794     L = { ,50},
8795     O = {25, },
8796     T = {50,50},
8797     V = {50,50},
8798     W = {50,50},
8799     X = {50,50},
8800     Y = {50,50},
8801     b = { ,25},
8802     d = {25,30},
8803     f = { ,50},
8804     g = { ,100},
8805     k = { ,50},
8806     p = { ,50},
8807     q = {50, },
8808     r = { ,50},
8809     t = { ,50}, ◆ = { ,50}, ◆ = { ,50},
8810     v = {75,50},
8811     w = {50,50},
8812     x = {50,50},
8813     y = {50,70},
8814     1 = {100,50},

```

```

8815 2 = {25,50},
8816 4 = {50, },
8817 6 = {50, },
8818 9 = {25, },
8819 Æ = {100, },
8820 Œ = {25, },
8821 . = { ,700}, .. = { ,350}, ... = { ,150},
8822 {,} = { ,500},
8823 := { ,500},
8824 ; = { ,500},
8825 ! = { ,100}, !! = { ,100},
8826 ? = { ,200}, ?? = { ,200},
8827 @ = {50,50},
8828 ~ = {200,250},
8829 & = {50,100},
8830 \% = {100,100},
8831 * = {200,200},
8832 + = {250,250},
8833 ( = {100, }, ) = { ,300},
8834 / = {200,300},
8835 - = {400,500},
8836 \textendash = {300,300}, \textemdash = {200,200},
8837 \textquoteleft = {500,700}, \textquoteright = {500,700},
8838 \textquotedblleft = {300,400}, \textquotedblright = {300,400},
8839 \textbackslash = {200,300},
8840 \quotingslbase = {400,400}, \quotedblbase = {400,400},
8841 \guillemotleft = {400,400}, \guillemotright = {300,500},
8842 \guillemotleft = {300,300}, \guillemotright = {200,400},
8843 \textexclamdown = {100, }, \textquestiondown = {100, },
8844 \textbraceleft = {400,200}, \textbraceright = {200,400},
8845 \textless = {200,100}, \textgreater = {100,200},
8846 ≤ = {200,100}, ≥ = {100,200},
8847 \textminus = {300,300},
8848 \texttrademark = {200,200},
8849 \textcopyright = {200,200},
8850 \textregistered = {200,200},
8851 \textdegree = {300,300},
8852 ¡ = {450,500}, ¬ = {250,150},
8853 ¯ = {150,250},
8854 · = {850,700},
8855 ¶ = {100,0},
8856 × = {150,300},
8857 ª = {300,300}, º = {300,300},
8858 ° = {200,400},
8859 ¹ = {400,350}, º = {200,300},  = {250,400},
8860 ¼ = {250,350}, ½ = {200,300}, ¾ = {250,400},
8861 ⅞ = {200,450}, ⅘ = {250,400}, ⅙ = {200,350},
8862 0 = {200,400},
8863 1 = {400,250}, 2 = {200,300}, 3 = {250,400},
8864 4 = {250,350}, 5 = {200,300}, 6 = {250,400},
8865 7 = {200,450}, 8 = {250,400}, 9 = {200,350},
8866 ± = {150,100}, ÷ = {300,300},
8867 þ = { ,25},
8868 ¸ = {300,450}, ˙ = {300,450},
8869 ˚ = {300,450}, ˛ = {300,450},
8870 † = {200,250}, ‡ = {200,250},
8871 π = {50, },
8872 f = { ,50},
8873 № = {100,150},
8874 \textservicemark = {100,200},
8875 - = {400,500}, - = {400,500}, - = {200,300},
8876 - = {205,305}, - = {200,300}, - = {50,150},
8877 • = {125,200},
8878 % /a.sc = {50,50},
8879 }

```

```

8880
8881 \SetProtrusion
8882 [ name = palatino-it ]
8883 { encoding = {EU1,EU2,TU},
8884   family = {PalatinoLinotype},
8885   shape = {it,sl} }
8886 {
8887 A = {50,50},
8888 Æ = {50, },
8889 B = {50, },
8890 C = {50, },
8891 D = {50,50},
8892 E = {50, },
8893 F = {50, },
8894 G = {50, },
8895 H = {50, },
8896 K = {50, },
8897 L = {50, },
8898 O = {50, },
8899 Œ = {50, },
8900 P = {50, },
8901 Q = {50, },
8902 R = {50, },
8903 S = {50, },
8904 $ = {50, },
8905 T = {100, },
8906 U = {50, },
8907 V = {100,50},
8908 W = {50, },
8909 X = {50, },
8910 Y = {100,50},
8911 b = { ,50},
8912 c = {25, },
8913 g = {75, },
8914 i = {25, },
8915 m = { ,50},
8916 n = { ,50},
8917 p = { ,25},
8918 q = {25, },
8919 x = { ,50},
8920 1 = {100, },
8921 2 = {50, },
8922 4 = {50, },
8923 7 = {50, },
8924 . = { ,500}, .. = { ,350}, ... = { ,200},
8925 {,} = { ,500},
8926 := { ,300},
8927 ; = { ,300},
8928 ? = { ,300}, ? = { ,300},
8929 & = {50,50},
8930 \% = {100,100},
8931 * = {200,200},
8932 + = {150,200},
8933 @ = {50,50},
8934 ~ = {200,150},
8935 ( = {200, }, ) = { ,200},
8936 / = {100,200},
8937 - = {300,500},
8938 \textendash = {300,300}, \textemdash = {200,200},
8939 \textquoteleft = {700,400}, \textquoteright = {700,400},
8940 \textquotedblleft = {500,300}, \textquotedblright = {500,300},
8941 _ = {100,100},
8942 \textbackslash = {100,200},
8943 \quotesinglbase = {500,500}, \quotedblbase = {400,400},
8944 \guilsinglleft = {400,400}, \guilsinglright = {300,500},

```

```

8945 \guillemotleft = {300,300}, \guillemotright = {300,300},
8946 \textexclamdown = {100, }, \textquestiondown = {200, },
8947 \textbraceleft = {200,100}, \textbraceright = {200,200},
8948 \textless = {300,100}, \textgreater = {200,100},
8949 ≤ = {200,100}, ≥ = {100,200},
8950 † = {450,500}, ‡ = {250,150},
8951 · = {850,700},
8952 ¶ = {100,0},
8953 × = {150,300},
8954 ª = {300,250}, ° = {300,300}, º = {300,250},
8955 º = {300,200},
8956 ¹ = {300,150}, ² = {350,200}, ³ = {250,150},
8957 ⁴ = {350,100}, ⁵ = {300,50}, ⁶ = {400,100},
8958 ⁷ = {400,50}, ⁸ = {250,50}, ⁹ = {300,50},
8959 ₀ = {300,300},
8960 ₁ = {300,350}, ₂ = {300,150}, ₃ = {250,250},
8961 ₄ = {400,200}, ₅ = {300,100}, ₆ = {450,200},
8962 ₇ = {450,150}, ₈ = {400,250}, ₉ = {400,200},
8963 ± = {150,100}, ÷ = {300,300},
8964 þ = {50, },
8965 † = {250,200}, ‡ = {250,200},
8966 ˙ = {300,450}, ˘ = {300,450},
8967 ˙ = {300,450}, ˘ = {300,450},
8968 - = {300,500}, - = {300,500}, - = {100,300},
8969 - = {125,305}, - = {200,300}, - = {125,150},
8970 • = {125,200}

8971 }
8972
8973 \SetProtrusion
8974 [ name = palatino-sc,
8975 load = palatino-default ]
8976 { encoding = {EU1,EU2,TU},
8977 family = {PalatinoLinotype},
8978 shape = sc }
8979 {

8980 a = {50,50},
8981 æ = {50, },
8982 b = {0,0},
8983 d = {0,0},
8984 f = {0,0},
8985 g = {0,0},
8986 j = {50, },
8987 l = { ,50},
8988 o = {0,0},
8989 p = {0,0},
8990 q = {0, },
8991 r = { ,0},
8992 t = {50,50},
8993 y = {50,50},
8994 fl = {0,50},
8995 ffl = {0,50},
8996 ◊ = {0,50},
8997 ◊ = {0,50}

8998 }
8999 /PalatinoLinotype
9000

```

17 Auxiliary file for micro fine tuning

This file can be used to test protrusion and expansion settings.

```

9001 (*test)
9002 \documentclass{article}
9003
9004 %% Here you can specify the font you want to test, using
9005 %% the commands \fontfamily, \fontseries and \fontshape.
9006 %% Make sure to end all lines with a comment character!
9007 \newcommand*\TestFont{%
9008   \fontfamily{ppl}%
9009   \fontseries{b}%
9010   \fontshape{it}% sc, sl
9011 }
9012
9013 \usepackage{ifthen}
9014 \usepackage[T1]{fontenc}
9015 \usepackage[latin1]{inputenc}
9016 \usepackage[verbose,expansion=alltext,stretch=50]{microtype}
9017
9018 \pagestyle{empty}
9019 \setlength{\parindent}{0pt}
9020 \newcommand\crulefill{\cleaders\hbox{$\mkern-2mu\smash-\mkern-2mu$}\hfill}
9021 \newcommand\testprotrusion[2][ ]{%
9022   \ifthenelse{\equal{#1}{r}}{\#2}%
9023   lorem ipsum dolor sit amet,
9024   \ifthenelse{\equal{#1}{r}}{\crulefill}{\leftarrowfill} #2
9025   \ifthenelse{\equal{#1}{l}}{\crulefill}{\rightarrowfill}
9026   you know the rest%
9027   \ifthenelse{\equal{#1}{l}}{\#2}%
9028   \linebreak
9029   {\fontencoding{\encodingdefault}%
9030   \fontseries{\seriesdefault}%
9031   \fontshape{\shapedefault}%
9032   \selectfont
9033   Here is the beginning of a line, \dotfill and here is its end}\linebreak
9034 }
9035 \newcommand\showTestFont{\expandafter\stripprefix\meaning\TestFont}
9036 \def\stripprefix#1>{}
9037 \newcount\charcount
9038 \begin{document}
9039
9040 \microtypesetup{expansion=false}
9041
9042 {\centering The font in this document is called by:\\
9043 \texttt{\showTestFont}\par}\bigskip
9044
9045 \TestFont\selectfont
9046 This line intentionally left empty\linebreak
9047 %% A -- Z
9048 \charcount=65
9049 \loop
9050   \testprotrusion{\char\charcount}
9051   \advance\charcount 1
9052   \ifnum\charcount < 91 \repeat
9053 %% a -- z
9054 \charcount=97
9055 \loop
9056   \testprotrusion{\char\charcount}
9057   \advance\charcount 1
9058   \ifnum\charcount < 123 \repeat
9059 %% 0 -- 9
9060 \charcount=48
9061 \loop

```



```

9062 \testprotrusion{\char\charcount}
9063 \advance\charcount 1
9064 \ifnum\charcount < 58 \repeat
9065 %%
9066 \testprotrusion[r]{,}
9067 \testprotrusion[r]{.}
9068 \testprotrusion[r]{;}
9069 \testprotrusion[r]{:}
9070 \testprotrusion[r]{?}
9071 \testprotrusion[r]{!}
9072 \testprotrusion[l]{\textexclamdown}
9073 \testprotrusion[l]{\textquestiondown}
9074 \testprotrusion[r]{\{ }
9075 \testprotrusion[l]{\{ (}
9076 \testprotrusion{/}
9077 \testprotrusion{\char~\}
9078 \testprotrusion{-}
9079 \testprotrusion{\textendash}
9080 \testprotrusion{\textemdash}
9081 \testprotrusion{\textquoteleft}
9082 \testprotrusion{\textquoteright}
9083 \testprotrusion{\textquotedblleft}
9084 \testprotrusion{\textquotedblright}
9085 \testprotrusion{\quotesinglbase}
9086 \testprotrusion{\quotedblbase}
9087 \testprotrusion{\guilsinglleft}
9088 \testprotrusion{\guilsinglright}
9089 \testprotrusion{\guillemotleft}
9090 \testprotrusion{\guillemotright}
9091
9092 \newpage
9093 The following displays the current font stretched by 5%,
9094 normal, and shrunk by 5%:
9095
9096 \bigskip
9097 \newlength{\MTln}
9098 \newcommand*\teststring
9099 {ABCDEFGHIJKLMNQRSTUvwxyz0123456789}
9100 \settowidth{\MTln}{\teststring}
9101 \microtypesetup{expansion=true}
9102
9103 \parbox{1.05\MTln}{\teststring\linebreak\}
9104 \parbox{0.95\MTln}{\teststring}\par\bigskip
9105 \parbox{0.95\MTln}{\teststring}
9106
9107 \end{document}
9108 /test

```

Needless to say that things may always be improved. For suggestions, mail to w.m.l@gmx.net.

A The title logo

This is `microtype-logo.dtx`. You may treat this file in three different ways:

- compile it by itself
- `\input` it in the body of a `dtx` file
- `\input` it in the preamble: it then provides the command `\printlogo`, which will do just that

The first two cases require the style file `microtype-doc.sty`, which can be generated from `microtype.ins` with:

```
\makefile{microtype-doc.sty}{docsty}
```

9109 *(*logo)*

Here's how the logo on the title page was created.³⁰ It has nothing to do with `microtype`, actually, but uses `fontinst`. It is based on an experiment I posted to the `de.comp.text.tex` newsgroup.³¹ It will show:

- the character
- the \TeX box
- the bounding box
- kerns

A.1 Macros

To run this file, \TeX needs to find the `afm` file (either in the `TEXINPUTS` path, or in the current working directory).

First input `fontinst`.

9110 `\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.

9111 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

9112 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

9113 `\newdimen\fboxrulei`

9114 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

9115 `\newdimen\fboxruleii`

9116 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

9117 `\newdimen\kernboxheight`

9118 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

9119 `\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).

9120 `\fontinstcc`

9121 `\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.

9122 `\ifdim\fontdimen6\font = 0pt`

9123 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%`

9124 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi-***}`

9125 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi\relax`

9126 `\fi`

9127 `\installfonts`

9128 `\input_metrics{}{\logofont,\metrics\printbbs{#1}\relax}`

30 Note that the `logo` module will not be created when installing `microtype`. Instead, the source file `microtype-logo.dtx` is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

31 Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

9129 \endinstallfonts
9130 }
9131 \normalcc
      Layers.
9132 \makeatletter
9133 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
9134 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
9135 \ifx\mt@order\undefined\let\mt@order\@empty\fi
9136 \xdef\mt@order{\mt@order[(Logo)]}
9137 \let\mtl@resources\@empty
9138 \def\mtl@register#1{%
9139   \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
9140   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdflastobj\space 0 R }
9141   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
9142   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
9143   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
9144 \mtl@register{canvas}
9145 \mtl@register{characters}
9146 \mtl@register{bounding-boxes}
9147 \mtl@register{TeX-boxes}
9148 \xdef\mt@order{\mt@order]}
9149 \global\let\mtl@objects\mt@objects
9150 \def\togglelayer#1#2{%
9151   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
9152   user{/Subtype/Link
9153     /BS << /Type/Border/W 0 >> /H/0
9154     /A << /S/SetOCGState
9155     /State[/Toggle \csname mtl@#1\endcsname] >>
9156   }#2\pdfendlink
9157 }

```

\printbbs Preparation.

```

9158 \setcommand\printbbs#1{%
9159   \setbox0\hbox{#1}%
9160   \leavevmode
9161   \kern-\fboxrulei
      The canvas in the natural width of the text minus protrusion, in color bgcolor.
9162   \mtl@layer{canvas}{%
9163     \getboundarychars#1\relax
9164     \tempdim=\dimexpr\wd0 - (\scalettoem{\lpcode\font\firstchar}+
9165       \scalettoem{\rpcode\font\lastchar})\relax
9166     \kern\dimexpr\scalettoem{\lpcode\font\firstchar}\relax
9167     \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
9168       \hrule width \tempdim
9169       height \dimexpr\dp0+\ht0+0.15em\relax}%
9170     \kern-\tempdim

```

The baseline, in color blcolor.

```

9171     \vbox{\color{blcolor}%
9172       \hrule width \tempdim
9173       height \fboxrulei}%
9174   }%
9175   \kern-\dimexpr\wd0 -\scalettoem{\rpcode\font\lastchar}\relax

```

The string.

```

9176   \printbbs #1\relax\relax
9177 }

```

\getboundarychars Get first

```

9178 \def\getboundarychars#1#2\relax{%
9179   \def\firstchar{~#1}%
9180   \getlastchar#1#2\relax
9181 }

```

\getlastchar ... and last character.

```

9182 \def\getlastchar#1#2{%

```

```

9183 \ifx\relax#2\relax
9184 \def\lastchar{`#1}%
9185 \else
9186 \expandafter\getlastchar
9187 \fi #2%
9188 }

```

`\printbss` Loop over all characters of the string.

```

9189 \def\printbss#1#2#3\relax{%
9190 \ifx\relax#1\relax
9191 \else
9192 \ifx\relax#2\relax
9193 \printbb{#1}{}%
9194 \else
9195 \printbb{#1}{#2}%
9196 \fi
9197 \expandafter\printbss
9198 \fi #2#3\relax
9199 }

```

`\printbb` Record the kern between the current and the following character, then print the character. `\kerning` is a fontinst command.

```

9200 \setcommand\printbb#1#2{%
9201 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
9202 \showboxes{#1}%

```

This could be another application.

```

9203 % \quad
9204 % w: \the\scaletoe{\width{#1}},
9205 % bb: \the\scaletoe{\bbleft{#1}}/%
9206 % \the\scaletoe{\bbright{#1}},
9207 % \the\scaletoe{\number\numexpr\width{#1}-\bbright{#1}\relax}
9208 % h: \height{#1}/\bbtop{#1}, \bbbottom{#1}/\depth{#1}\par
9209 }

```

`\showboxes` Print the boxes for char `<#1>`. This won't work if `<#1>` isn't also the PostScript name of the glyph (e.g., 'comma' ≠ ',').

```

9210 \setcommand\showboxes#1{%
9211 \leavevmode
9212 \color{texcolor}%

```

We have to record the width of the glyph.

```

9213 \setbox0\hbox{\color{textcolor}{#1}}%
9214 \global\tempdim=\wd0\relax
9215 \kern-\fboxrulei

```

1. *The TeX box*: Print a frame in color `texcolor`. This frame shows the glyph as TeX sees it.

```

9216 \mtl@layer{TeX-boxes}{%
9217 \hbox{%
9218 \lower\dimexpr \dp0 + \fboxrulei\relax
9219 \hbox{%
9220 \vbox{%
9221 \hrule height\fboxrulei
9222 \hbox{%
9223 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
9224 \phantom{\unhcopy0}%
9225 \vrule width\fboxrulei
9226 }%
9227 \hrule height\fboxrulei}}}%
9228 }%

```

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.

```

9229 \kern-\wd0
9230 \mtl@layer{characters}{\hbox{\box0}}%

```

Step back by the amount that the character's bounding box differs from the TeX box on the left side.

```

9231 \kern\dimexpr\scaletoe{\bbleft{#1}}-\tempdim-\fboxrulei\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

9232 \mtl@layer{bounding-boxes}{%
9233   {\color{bbcolor}%
9234   \hbox{%
9235     \lower\dimexpr-\scaletoe{\bbottom{#1}}+\fboxrulei\relax
9236     \hbox{%
9237       \vbox{%
9238         \hrule height\fboxrulei
9239         \hbox to \dimexpr\scaletoe{\numexpr
9240           \bbright{#1}-\bbleft{#1}\relax}+2\fboxrulei\relax{%
9241           \vrule height \dimexpr\scaletoe{\numexpr
9242             \bbtop{#1}-\bbottom{#1}\relax}%
9243             width\fboxrulei
9244             \hfill
9245             \vrule width\fboxrulei}%
9246             \hrule height\fboxrulei}}}%
9247     }%
9248     \kern-\dimexpr\fboxrulei+\fboxrulei\relax
9249   }%

```

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.

```

9250 \kern\scaletoe{\numexpr\width{#1}-\bbright{#1}\relax}%
9251 \mtl@layer{TeX-boxes}{%
9252   {\ifnum\thekern<0
9253     \color{kerncolor}%
9254     \kern\scaletoe{\thekern}%
9255     \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletoe{\thekern}\relax
9256       height \kernboxheight}%
9257     \kern\scaletoe{\thekern}%
9258   }else
9259     \color{texcolor}%
9260     \ifnum\thekern=0 \else
9261       \lower\kernboxheight
9262       \hbox{%
9263         \vbox{%
9264           \hrule height\fboxrulei
9265           \hbox{%
9266             \vrule height \kernboxheight width\fboxrulei
9267             \kern\dimexpr\scaletoe{\thekern}-2\fboxrulei\relax
9268             \vrule width\fboxrulei
9269           }%
9270           \hrule height\fboxrulei}}}%
9271       \fi
9272     \fi
9273   }%
9274 }%
9275 % \kern-\fboxrulei
9276 }

```

```

9277 \newbox\logobox
9278 \def\printlogo{%
9279   \setbox\logobox=\hbox{\vbox{%
9280     \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

9281 \def\logofont{pkpri9e10}
9282 \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
9283 \font\thelogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

9284 %\def\logofont{pplri}
9285 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
9286 %\edef\logofont{\logofont8r}
9287 %\font\thelogofont=\logofont\space at 78pt

```

Load the font.

```

9288 \thelogo font
      Protrusion values (overdone for didactic reasons).
9289 \lcode\font`M=96
9290 \rcode\font`e=46

      Now we can generate the logo.
9291 \pdfliteral direct{/SXS gs}%
9292 \showlogo{Microtype}%
9293 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
9294 % \kern5pt\[\[3\baselineskip]
9295 % \long\def\@makefnmark##1{%
9296 % \leftskip 0pt
9297 % \parindent 0pt
9298 % \everypar{\parindent 0pt}%
9299 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
9300 % \footnotetext[1]{This graphic display on a
9301 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
9302 % their \togglelayer{bounding-boxes}{bounding boxes}
9303 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.}
9304 % }}%
9305 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
9306 \immediate\pdfobj{<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>}%
9307 \immediate\pdfxform
9308     attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
9309     resources {/Properties <<\mtl@resources>>
9310               /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
9311     \logobox
9312 % \vskip-2.5\baselineskip
9313 % \leavevmode
9314 % \togglelayer{characters}{%
9315 % \pdfrefxform\pdflastxform
9316 % }%
9317 \pdfannot\logodimens{%
9318     /Subtype/Widget /FT/Btn /T(Logo)
9319     %/F 4 % why did I say this?
9320     /AP << /N \the\pdflastxform\space 0 R >>
9321     /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
9322         /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
9323         /D << /S/SetOCGState /State[/Toggle \csname mtl@bounding-boxes\endcsname] >>
9324         /U << /S/SetOCGState /State[/Toggle \csname mtl@TeX-boxes\endcsname] >>
9325         >> }%
9326 \vspace{3\baselineskip}
9327 }

      Our font.
9328 \pdfmapline{+pkpmmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmmri8a10.pfb}
      Define colours (thered and thegreen are copied from microtype.dtx).
9329 \def\mtdefinecolors{
9330 \definecolor{thered}{rgb}{0.65,0.04,0.07}
9331 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
9332 \colorlet{texcolor}{thegreen!50} % TeX boxes
9333 \colorlet{kerncolor}{texcolor} % negative kerns
9334 \colorlet{bbcolor}{thered!50} % bounding box
9335 \colorlet{bgcolor}{black!8} % canvas
9336 \colorlet{blcolor}{black!50} % baseline
9337 \colorlet{textcolor}{black!40} % text
9338 }

      Use with microtype.dtx
9339 \ifx\documentclass\@twoclasseserror
9340 \usepackage{xcdraw}{xcolor}
9341 \mtdefinecolors
9342 \else

```

A.2 Document

Now we can start the document.

```

9343 \documentclass[10pt,a4paper]{ltxdoc}
9344 \providecommand\MakePercentComment{\relax}
9345 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}
    Re-use the preamble from microtype.dtx.
9346 \usepackage{microtype-doc}
9347 \usepackage{attachfile}
9348 \makeatletter
9349 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
9350 \makeatother
9351 \begin{document}
    You are currently reading this.
9352 \DocInput{microtype-logo.dtx}
9353 \newpage
9354 And here it is:
9355 \vfill
9356 \begin{center}
9357 \printlogo \null
9358 \end{center}
9359 \vfill
9360 \expandafter\enddocument
9361 \fi
    That's it.
9362 /Logo

```

B The letterspacing illustration

This is `microtype-1ssample.dtx`. You may treat this file in three different ways:

- compile it by itself
- `\input` it in the body of a dtx file
- `\input` it in the preamble: it then provides the commands
 - `\1ssample`: prints the letterspacing illustration
 - `\anchorarrow`: anchors an arrow for layer `<#1>`
 - `\showarrow`: toggles layer `<#1>` or `<#2>`, and prints `<#2>`

The first two cases require the style file `microtype-doc.sty`, which can be generated from `microtype.ins` with:

```
\makefile{microtype-doc.sty}{docsty}
```

```

9363 \ifx\1ssample\undefined
9364 *1ssample

```

Upon popular request, here's how I've created the letterspacing illustration.³²

B.1 Macros

Rule width and image height and depth.

```

9365 \makeatletter
9366 \newdimen\1samount
9367 \newdimen\1srule
9368 \1srule=0.2pt
9369 \def\1sheight{8pt}
9370 \def\1sdepth{12pt}

```

³² Note that the `1ssample` module will not be created when installing `microtype`. Instead, the source file `microtype-1ssample.dtx` is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

Our font (Adobe Caslon).

```

9371 \def\lsfont{\fontfamily{paca}\selectfont}
      Loop over all letters in <#2>, letterspacing them by <#1>.
9372 \def\dols#1#2{\lssamount=#1\relax \dolss#2\enddols}
9373 \def\dolss#1#2\enddols{%
9374   \ifx\empty#2\empty\divide\lssamount 2\fi
9375   \ls{#1}%
9376   \ifx\empty#2\empty\else \dolss#2\enddols \fi
9377 }

```

One tikz picture for each letter.

```

9378 \def\ls#1{%
9379   \begin{tikzpicture}[remember picture,line width=\lsrule]
9380     \tikzstyle{every node}=[inner sep=0pt]

```

The bounding box.

```

9381     \mts@layer{stuff}{%
9382       \node[draw=thegrey,
9383         fill=theshade,
9384         outer sep=\lsrule,
9385         anchor=base,
9386         font=\lsfont]{\phantom{#1}};
9387     }

```

The letter.

```

9388     \node[anchor=base,font=\lsfont](#1){#1};

```

Two auxiliary coordinates.

```

9389     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
9390     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
9391     \mts@layer{stuff}{%

```

Now draw the normal character width,

```

9392       \draw[color=thered!75,
9393         fill=thered!30,
9394         outer sep=\lsrule]
9395         (#1L) rectangle (#1R);
9396       \ifdim\lssamount>0pt
9397         \path (#1.base east) ++(+.5\lssamount,-6pt) coordinate (#1_1s);
9398         \path (#1R) ++(\lssamount+\lsrule,\lsdepth) coordinate (#1E);

```

and the letter space.

```

9399       \draw[color=thered,
9400         fill=thered!50,
9401         outer sep=\lsrule]
9402         (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
9403     \fi
9404   }
9405 \end{tikzpicture}%
9406 \ignorespaces
9407 }

```

Draw the interword space.

```

9408 \def\lssp#1#2#3#4{%
9409   \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
9410     \mts@layer{stuff}{%
9411       \tikzstyle{every draw}=[anchor=bottom]
9412       \coordinate(#1space) at (#2/2,\lsdepth/2);
9413       \coordinate(#1stretch) at (#2+#3/2,+0pt);
9414       \coordinate(#1shrink) at (#2-#4/2,+0pt);
9415       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
9416         (0,0) rectangle ++(#2,\lsdepth);
9417       \draw[color=thegreen,fill=thegreen!30]
9418         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
9419       \draw[color=thegreen,fill=thegreen!50]
9420         (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
9421       \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]

```



```

9422         (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
9423     \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
9424         (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
9425     }%
9426 \end{tikzpicture}%
9427 \ignorespaces
9428 }

Layers.
9429 \def\mts@layer#1#2{\pdfliteral page{/OC/#1 BDC}#2\pdfliteral page{EMC}}
9430 \def\mts@layer#1#2{\pdfliteral page{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral page{EMC EMC}}
9431 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
9432 \ifx\mt@order \undefined\let\mt@order \@empty\fi
9433 \xdef\mt@order{\mt@order[(Sheep)]}
9434 \let\mts@resources\@empty
9435 \def\mts@register#1{%
9436     \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
9437     \expandafter\xdef\csname mts@#1\endcsname{\the\pdfastobj\space 0 R }
9438     \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
9439     \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
9440     \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
9441 \mts@register{stuff}
9442 \mts@register{tracking}
9443 \mts@register{ispace}
9444 \mts@register{ospace}
9445 \mts@register{istretch}
9446 \mts@register{ishrink}
9447 \mts@register{ostretch}
9448 \mts@register{oshrink}
9449 \mts@register{okern}
9450 \mts@register{ligature}
9451 \mts@register{_compatibility}
9452 \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
9453 \newcommand\anchorarrow[1]{%
9454     \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
9455 \newcommand\add@arrow[5][left]{%
9456     \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
9457         \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}%
9458     }

Toggle layer.
9459 \def\toggle@layer#1#2#3{%
9460     \pdfstartlink
9461     user{/Subtype/Link
9462         /BS << /Type/Border/W 0 >> /H/0
9463     %     /BS << /Type/Border/W 1 /S/D /D[4 1] >>
9464     %     /C[0.7 0.7 0.7] /H/0
9465         /Contents(Click to Toggle!)
9466         /A << /S/SetOCGState
9467             /State[/Toggle \csname mts@#1\endcsname] >> }%
9468     \rlap{#2}%
9469     {\fboxsep=0pt \fboxrule=0pt
9470     \mts@layer{stuff}{%
9471         \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
9472     \mts@layer{#1}{%
9473         \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}}%
9474     }%
9475     \pdfendlink
9476     }
9477 \newcommand\showarrow[2][ ]{%
9478     \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
9479     \toggle@layer{\@tempa}{\itshape #2}}

```

The environment for our illustration.

```

9480 \def\ls@sample#1{%
9481   \parskip 4pt \parindent 0pt
9482   \par
9483   \vskip4pt
9484   {\leftskip 15pt
9485    \mt@pseudo@margin{\color{theblue}Click on the image to show the kerns
9486     and spacings involved. Click on emphasised words in the text below
9487     to reveal the relation of image and code.\strut}
9488    \mt@layer{_compatibility}{%
9489     \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
9490      \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
9491     \mt@pseudo@margin{\color{thered}%
9492      If you had a \acronym{PDF} viewer that understands
9493      \acronym{PDF}\,\{\smaller1.5}, you could hide the arrows selectively.}}
9494     \vskip-\mt@unvdimen}%
9495   \vskip-4pt
9496   \setlength\fbxsep{4pt}%
9497   \leavevmode
9498   \pdfstartlink
9499     user{/Subtype/Link
9500       /BS << /Type/Border/W 0 >> /H/0
9501       /A << /S/SetOCGState
9502         /State[/Toggle \mts@stuff] >> }%
9503     \fcolorbox{theframe}{theshade}%
9504     {\fontsize{34}{38}\selectfont #1}%
9505   \pdfendlink
9506   \par\medskip
9507   }%
9508   \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
9509 }

```

Now define the illustration to be used in the document.

```

9510 \def\lssample{%
9511   \ls@sample{%
9512     \dols{Opt}{Stop}
9513     \lssp{o}{0.45em}{0.25em}{0.15em}
9514     \dols{0.16em}{\st ealing}\hskip-\dimexpr 0.08em+\lsrule\relax
9515     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
9516     \dols{0.16em}{sheep}
9517     \dols{Opt}{!}
9518   }%

```

Don't forget to add the arrows.

```

9519   \vspace{-\baselineskip}
9520   \add@arrow{red}      {tracking}{lsamount_c.east}{a_ls}
9521   \add@arrow{red}      {okern}   {okernend_c.east}{p_ls}
9522   \add@arrow{green}    {ospace}  {ospace_c.east}  {ospace}
9523   \add@arrow{green}    {ispace}  {ispace_c.center}{ispace}
9524   \add@arrow{green!75} {istretch}{istretch_c.east}{istretch.north}
9525   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
9526   \add@arrow{green!75} {ostretch}{ostretch_c.east}{ostretch.north}
9527   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
9528   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
9529 }
9530 \fi

```

This is for use with microtype.dtx

```

9531 \ifx\documentclass\@twoclasseserror
9532   \usepackage{tikz}
9533 \else

```

B.2 Document

```

9534 \documentclass[10pt,a4paper]{ltxdoc}
9535 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

```

```

Re-use the preamble from microtype.dtx.
9536 \usepackage{microtype-doc}
9537 \usepackage{attachfile}
9538 \usepackage{tikz}
9539 \makeatletter
9540 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
9541                               /D << /Order [\mt@order] /BaseState/OFF >> >> }
9542 \makeatother
9543 \begin{document}
  You are currently reading this.
9544 \DocInput{microtype-lssample.dtx}
  Now show what we are able to do.
9545 \noindent
9546 Since a picture is worth a thousand words, probably even more if, in our
9547 case, it depicts a couple of letterspaced words, let's bring one to sum up
9548 these somewhat confusing options. Suppose you had the following settings
9549 (which I would in no way recommend; they are only for illustrative purposes):
9550 \begin{verbatim}
9551 \SetTracking
9552 [ no ligatures = {"\anchorarrow{nolig}"f},
9553   spacing      = {60"\anchorarrow{ispace}"0*, "%
9554                  -1"\anchorarrow{istretch}"00*, "\anchorarrow{ishrink}"},
9555   outer spacing = {4"\anchorarrow{ospace}"50, "%
9556                  2"\anchorarrow{ostretch}"50, 1"\anchorarrow{oshrink}"50},
9557   outer kerning = {"\anchorarrow{okernbegin}"*, "%
9558                  \anchorarrow{okernend}"* } ]
9559 { encoding = * }
9560 { 1"\anchorarrow{lsamount}"60 }
9561 \end{verbatim}
9562 and then write:
9563 \begin{verbatim}
9564 Stop \textls{stealing sheep}!
9565 \end{verbatim}
9566 this is the (typographically dubious) outcome:
9567
9568 \lssample
9569
9570 \noindent
9571 While the word 'Stop' is not letterspaced, the space between the letters in
9572 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
9573 of 160/1000\,em\,=\allowbreak\,0.16\,em.
9574 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
9575 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
9576 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
9577 untouched.
9578 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
9579 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
9580 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
9581 Note that there is no outer space after the text, since the exclamation mark
9582 immediately follows; instead, the default \showarrow[okern]{outer~kern}{red}
9583 of half the letterspace amount (0.08\,em) is added.
9584 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
9585 neglected to specify the '~|s|' in the |no ligatures| key.
9586
9587 \expandafter\enddocument
9588 \fi
9589 </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*) 87
 remove 8-bit characters from the configuration files (suggested by *Harald Harders*) 146
 Protrusion: add factors for some more characters 153
 settings for Adobe Minion (contributed by *Harald Harders*) 154
 \backslash DeclareCharacterInheritance: new command: possibility to specify character inheritance 119
 \backslash MT@declare@sets: remove spaces around set name 105
 \backslash MT@find@file: fix: also check whether the file for the base font family has already been loaded . . 87
 \backslash MT@get@basefamily: only remove suffixes ‘x’ or ‘j’ 88
 \backslash MT@get@listname@: don’t check for empty attributes list 89
 \backslash MT@ifempty: fix: use category code 12 for the percent character (reported by *Tom Kink*) 45
 \backslash MT@is@number: numbers may also be specified in hexadecimal or octal (suggested by *Harald Harders*) 94
 \backslash MT@pdftex@no: fix: version check (reported by *Harald Harders*) 40
 \backslash MT@permute: don’t use sets for empty encoding . . . 121
 \backslash MT@setup@expansion: issue an error instead of a warning, when pdfTeX version is too old for autoexpand 136
 \backslash MT@split@codes: fix: allow zero and negative values 63
 \backslash MT@use@set: remove spaces around set name . . . 109

2004/10/03 **Version 1.2**

Font aliases: declare cmor as an alias of cmr 143
 Font sets: new: allmath and basicmath 142
 Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding 177
 add settings for Computer Modern Roman math symbols 182
 \backslash MT@familyalias: define alias font name as an alternative, not as a replacement 59
 \backslash MT@get@basefamily: also remove ‘w’ (swash capitals) 88
 \backslash MT@get@highlevel: check whether defaults have changed 106
 \backslash MT@get@inh@list: fix: set inheritance list \globally to \empty 91
 \backslash MT@get@listname@: alternatively check for alias font name 89
 \backslash MT@get@size: additional magic to catch some errors hijack \set@fontsize instead of \@setfontsize 107
 \backslash MT@loop: fix: new macro, used instead of \loop . . 49
 \backslash MT@maybe@do: also check for alias font name 60
 \backslash MT@permute@@@@: more sanity checks for \SetProtrusion and \SetExpansion 122
 \backslash MT@setupfont: also search for alias font file 57
 fix: call \@@enc@update if necessary 57

2004/10/27 **Version 1.3**

General: fix: specifying load option does no longer require to give a name, too 116
 Font aliases: declare aer, zer and hfor as aliases of cmr 143
 \backslash MT@fix@catcode: check some category codes (compatibility with german) 35
 \backslash MT@load@list: check whether list exists 87

2004/11/12 **Version 1.4**

General: check for pdfcprot 54
 don’t use scratch registers in global definitions . . 91
 use \pickup@font instead of \define@newfont as the hook for \MT@setupfont 99
 use one instead of five counters 50
 Protrusion: tweak quote characters for cmr variants (OT1, T1, lmr) 158
 \backslash microtypesetup: fix: set the correct levels, and remember them; warning when enabling an option disabled in package options 130
 \backslash SetExpansion: fix: specifying extra options does no longer require to give a name, too 113

2004/11/17 **Version 1.4a**

General: new option: final 127
 \backslash MT@cfg@catcodes: fix: reset some more catcodes when reading files (reported by *Michael Hoppe*) 88

2004/11/26 **Version 1.4b**

General: fix: set catcodes before reading global configuration file (reported by <i>Christoph Bier</i>) . . .	129	form abczz (reported by <i>Georg Verweyen</i>)	88
optimisation: use less <code>\expandafers</code> and <code>\csnames</code>	44	<code>\MT@get@slot</code> : don't define <code>\MT@char</code> globally (save stack problem)	91
Protrusion: harmonise dashes in upshape and italic (cmr, pad, ppl)	153	<code>\MT@ifdimen</code> : don't set <code>\MT@count</code> globally (save stack problem)	46
slanted like italics	162	<code>\MT@setup@PDF</code> : new message if <code>\pdfoutput</code> is changed	133
<code>\MT@checklist@family</code> : fix: don't try alias family name if encoding failed	61	<code>\MT@use@set</code> : don't use undeclared font sets	109
<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>)	127	<code>\MT@get@highlevel</code> : don't test defaults if called after begin document	106
new option: selected, by default false (suggested by <i>Hàn Thế Thành</i>)	125	<code>\MT@scale@factor</code> : warning for factors outside limits	65
Documentation: add 'Short history'	30	<code>\MT@scale@to@em</code> : don't use <code>\lcode</code> and <code>\rcode</code> for the calculation	64
Inheritance: remove <code>\ss</code> from T1 list, add <code>\DJ</code> . . .	146	<code>\MT@set@ex@codes</code> : allow non-selected font expansion	69
Protrusion: settings for Bitstream Charter	154	<code>\MT@set@pr@codes</code> : adjust protrusion factors before setting the inheriting characters	62
<code>\DeclareMicrotypeAlias</code> : remove spaces around arguments	110	<code>\MT@setup@expansion</code> : defaults: calculate step as $\min(\text{stretch}, \text{shrink})/5$	135
<code>\MT@cfg@catcodes</code> : reset catcode of <code>'=</code> ' (compatibility with Turkish babel)	88	defaults: turn off expansion for DVI output . . .	134
<code>\MT@fix@catcode</code> : reset catcode of <code>'^</code> ' (compatibility with chemsym)	35	disable automatic expansion for DVI output . . .	135

2005/01/24 **Version 1.6**

General: defaults: turn off expansion for old pdfTeX versions	128	tune CMR math letters (OML encoding)	182
load a font if none is selected	56	<code>\MT@get@charwd</code> : use e-TeX's <code>\fontcharwd</code> , if available	64
new option: factor, by default 1000	127	<code>\MT@get@inh@list</code> : correct message if selected is false	90
restructure dtx file	142	<code>\MT@set@ex@codes</code> : introduce factor option	69
test whether <code>\pickup@font</code> has changed	102	<code>\MT@set@pr@codes</code> : introduce factor option	62
test whether numeric options receive a number	127	<code>\MT@setup@expansion</code> : disable automatic expansion for old pdfTeX versions	136
use e-TeX's <code>\ifcsize</code> and <code>\ifdefined</code> if defined	44	<code>\MT@use@set</code> : retain current set if new set is undeclared	109
Protrusion: add italic uppercase Greek letters . . .	162	<code>\MT@vinfo</code> : new macro instead of <code>\ifMT@verbose</code> . .	36
improve settings for numbers (pointed out by <i>Peter Muthesius</i>)	155		

2005/02/02 **Version 1.6a**

Documentation: add table of fonts with tailored protrusion settings	21	reported by <i>Bernard Gaulle</i>)	91
<code>\MT@get@slot</code> : completely redone, hopefully more robust (compatible with frenchpro; problem		<code>\MT@pdfTeX@no</code> : new macro	39
		<code>\MT@reset@ef@codes</code> : only reset <code>\efcodes</code> for older pdfTeX versions	70

2005/03/23 **Version 1.7**

General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i>)	106	<code>\textbackslash</code> to T1 encoding	156
disallow automatic expansion if pdfTeX too old	118	<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i>)	111
fix: remove space after <code>autoexpand</code>	118	<code>\Microtype@Hook</code> : new command for font package authors	129
new value for verbose option: errors	127	<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	130
shorter command names	51	<code>\MT@begin@catcodes</code> : also use inside configuration commands	88
warning when running in draft mode	133	<code>\MT@cfg@catcodes</code> : reset catcode of <code>':</code> ' (compatibility with french* packages)	88
Documentation: add hint about compatibility . . .	26		
remove table of match order (now table 4 on page 89)	12		
Protrusion: fix: remove <code>\</code> from OT1, add			

<code>\MT@DeclareMicrotypeAlias</code> : may also be used inside configuration files	111	<code>\MT@scale</code> : new macro: use e-TeX's <code>\numexpr</code> if available	51
<code>\MT@get@listname@</code> : use <code>\@tfor</code> (<i>Andreas Bühmann's</i> idea)	89	<code>\MT@set@ex@codes</code> : two versions of this macro	69
<code>\MT@get@slot</code> : remove backslash hack	91	<code>\MT@split@name</code> : don't define <code>\MT@encoding</code> &c. globally	59
test for <code>\chardefed</code> commands	92	<code>\MT@test@ast</code> : make it simpler	106
test whether <code>\(encoding)\{...}</code> is defined	91	<code>\MT@try@order</code> : always check for size, too (suggested by <i>Andreas Bühmann</i>)	89
<code>\MT@if@list@exists</code> : don't define <code>\MT@#1@cname</code> globally, here and elsewhere	90	fix: also check for <code>//(series)/(shape)//</code> (reported by <i>Andreas Bühmann</i>)	89
<code>\MT@if@dimen</code> : comparison with 1 to allow size smaller than 1 (suggested by <i>Andreas Bühmann</i>)	46	<code>\MT@warn@code@too@large</code> : new macro: type out maximum protrusion factor	66
<code>\MT@increment</code> : use e-TeX's <code>\numexpr</code> if available	50	<code>\MT@warn@err</code> : new macro: for <code>verbose=errors</code>	36
<code>\MT@is@composite</code> : new macro: construct command for composite character; no uncontrolled expansion	97	<code>\showhyphens</code> : modify <code>\showhyphens</code>	137

2005/06/23 **Version 1.8**

General: <code>\SetProtrusion</code> : new key: unit	117	<code>\MT@find@file</code> : no longer wrap names in commands	87
if font substitution has occurred, set up the substitute font, not the selected one	99	<code>\MT@fix@fontdimen@six</code> : new macro: test whether <code>\fontdimen6</code> is defined	59
new option: config to load a different main configuration file	128	<code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters	64
new option: unit, by default character	128	<code>\MT@get@listname@</code> : made recursive	89
Documentation: add example for factor option	13	<code>\MT@get@slot</code> : fix: expand active characters	91
add example of how to get rid of a widow (suggested by <i>Adam Kucharczyk</i>)	15	test whether <code>\(encoding)\{...}</code> is defined made more robust	91
add hint about error messages	27	<code>\MT@get@unit</code> : new macro: get unit for codes	67
Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ppl</code> resp. <code>ptm</code>	144	<code>\MT@in@rlist</code> : made recursive	49
Font sets: add U encoding to <code>allmath</code>	142	<code>\MT@is@active</code> : new macro: translate inputenc-defined characters	95
Inheritance: remove <code>\DJ</code> from T1 list (it's the same as <code>\DH</code>)	146	<code>\MT@is@letter</code> : warning for non-ASCII characters	94
Protrusion: add LY1 characters for Times	161	<code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code>	52
settings for AMS math fonts	185	<code>\MT@map@clist@n</code> : new macro: used instead of <code>\@tfor</code>	48
verified settings for slanted Computer Modern Roman	170	<code>\MT@map@tlist@n</code> : new macro: used instead of <code>\@tfor</code>	48
<code>\add@accent</code> : fix: disable micro-typographic setup inside <code>\add@accent</code> (reported by <i>Stephan Hennig</i>)	101	<code>\MT@old@cmd</code> : renamed commands from <code>\..MicroType..</code> to <code>\..Microtype..</code>	36
<code>\DeclareMicrotypeAlias</code> : warning when overriding an alias font	110	<code>\MT@pdftex@no</code> : case 5: pdfTeX 1.30	40
<code>\DeclareMicrotypeSetDefault</code> : new command: set default font set	109	<code>\MT@permute@@@@</code> : add ranges to the beginning of the lists	122
<code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters	88	<code>\MT@scale</code> : fix: remove spaces in e-TeX variant (reported by <i>Mark Rossi</i>)	51
<code>\MT@check@rlist</code> : made recursive	123	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded	54
<code>\MT@curr@list@name</code> : new macro: current list type and name	98	restore <code>csquotes's</code> active characters	54
<code>\MT@declare@sets</code> : warning when redefining a set	105	restore percent character if Spanish <code>babel</code> is loaded	54
<code>\MT@define@set@key@</code> : use comma lists instead of token lists	105	<code>\MT@split@codes</code> : get character width once only	63
		<code>\MT@use@set</code> : fix: remove braces in first line	109
		<code>\MT@xadd</code> : simplified	47

2005/10/28 **Version 1.9**

General: <code>\DeclareMicrotypeSet</code> : new key: font	108	Documentation: add hint about <code>verbatim</code> environment	25
<code>\SetProtrusion</code> : value 'relative' renamed to 'character' for key unit	117	add remark about Type 1 fonts required for automatic font expansion	8
allow context-specific font setup	99	Font aliases: declare <code>qpl</code> and <code>qtm</code> (qfonts, TeX Gyre) as aliases of <code>ppl</code> resp. <code>ptm</code>	144
compatibility with TeX Live hack (reported by <i>Herrbert Voß</i>)	39	Font sets: add OT4 encoding to text sets	142
disable microtype setup inside <code>hyperref's</code> <code>\pdfstringdef</code> (reported by <i>Hàn Thế Thành</i>)	55	add T5 encoding to text sets	142
fix: use <code>true</code> as the default value	124	Inheritance: add list for OT4	148
option unit: rename value relative to character	128	add list for T5 (requested by <i>Hàn Thế Thành</i>)	149

Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR	157	<code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists	67
settings for OT4 encoding (Computer Modern Roman, Palatino, Times)	153	<code>\MT@is@active</code> : redone: use <code>\set@display@protect</code>	95
settings for T5 encoded Computer Modern Roman	153	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code>	94
<code>\DisableLigatures</code> : new command: disable ligatures (requires pdfTeX 1.30)	111	<code>\MT@maybe@do</code> : redone	60
<code>\microtypecontext</code> : new command: change setup context in the document	103	<code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list	48
<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code>	61	<code>\MT@scale@factor</code> : generalised	65
<code>\MT@detokenize@c</code> : fix the \LaTeX version	45	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero	136
<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code>	44	warning if user requested zero step	135
		<code>\MT@toks</code> : use instead of <code>\toks@</code>	42
		<code>\SetProtrusion</code> : (et al.) new key: font	112

2005/12/05 **Version 1.9a**

General: ‘ <code>{file name}/(line number)</code> ’ as default list name	116	diately (requested by <i>Georg Verweyen</i>)	105
new option: <code>defersetup</code> , by default true	126	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed	106
remove superfluous test whether <code>\pickup@font</code> has changed	102	<code>\MT@ifdefined@cc@T</code> : new macros: true case only	44
Documentation: add explanation for error message in DVI mode	27	<code>\MT@ifint</code> : use <code>\pdfmatch</code> if available	45
add explanation for error message with non-Type 1 fonts	28	<code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available	47
Font aliases: declare <code>mbch</code> (<code>mathdesign</code>) as an alias of <code>bch</code>	145	<code>\MT@in@clist</code> : fix	48
Protrusion: fix: remove ‘ <code>_</code> ’ from OT1 encoding	157	<code>\MT@info@missing@char</code> : info instead of warning is missing in Palatino SC)	65
settings for T5 encoded Charter	153	<code>\MT@is@feature</code> : new macro: check for pdfTeX feature	51
<code>\microtypesetup</code> : inside the preamble, accepts all package options	130	<code>\MT@map@clist@n</code> : following \LaTeX 3	48
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	102	<code>\MT@permute@@@</code> : don’t define permutations for unused encodings	121
<code>\MT@define@set@key@</code> : don’t expand variables immediately		<code>\MT@rem@from@clist</code> : fix	48
		<code>\MT@setup@</code> : defer setup until the end of the preamble	52

2006/01/20 **Version 1.9b**

General: compatibility with <code>listings</code> : sanitise more catcodes (reported by <i>Holger Uhr</i>)	55	add samples of micro-typographic features	4
compatibility with the <code>extendedchar</code> option of the <code>listings</code> package	55	<code>\MT@features</code> : use throughout the package to adjust to beta-ness	51
Documentation: activate expansion in the distributed PDF	1	<code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available	46
		<code>\MT@warn@code@too@large</code> : fix calculation with present factor	66

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	115
Protrusion: settings for URW Garamond	154	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b)	115

2006/05/05 **Version 1.9d**

Font sets: <code>md*</code> instead of <code>m</code> series in basic sets	142	<code>\DeclareCharacterInheritance</code> : fix: empty context	119
add QX encoding to text sets	142	<code>\MT@detokenize@n</code> : new macro: use <code>\detokenize</code> if available	45
Inheritance: add list for QX encoding (contributed by <i>Maciej Eder</i>)	148	<code>\MT@get@ex@opt</code> : fix: evaluate preset	71
Protrusion: settings for QX encoding (contributed by <i>Maciej Eder</i>)	160	<code>\MT@get@font@dimen</code> : warning for zero fontdimen	65
settings for Euro symbols (Adobe, ITC, marvosym)	193	<code>\MT@get@opt</code> : optimise: don’t reset when preset option is set	67
tweak AMS settings	185	set list name before presetting	67

<code>\MT@is@active</code> : support for Unicode (inputenc/utf8)	95	mandatory argument	112
<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when tex4ht is loaded (reported by <i>Peter Dyballa</i>)	54	(et al.) split keys of optional and mandatory argument	112
<code>\SetProtrusion</code> : (et al.) optimise: unify keys for			

2006/07/28 **Version 1.9e**

General: fix: default value for <code>activate: true</code>	125	settings for Euler Roman font	189
Documentation: add hint about unknown encodings include LPL	26 246	<code>\DeclareCharacterInheritance</code> : new key ‘inputenc’ to set the input encoding	119
Font aliases: declare <code>zeur</code> and <code>zeus</code> (eulervm) as aliases of <code>eur</code> resp. <code>eus</code> (euler)	145	<code>\MT@rem@from@clist</code> : model after <code>\@removeelement</code>	48
Inheritance: adapt to marvosym’s changed encoding	150	<code>\MT@setup@</code> : empty <code>\MT@setup@</code> after use (compatibility with the <code>combine</code> class)	52
Protrusion: complete settings for Euler Fraktur and Script fonts	192	<code>\pickup@font</code> : no tracing with <code>trace</code> package	101
fix: forgotten comma in <code>mt-mvs.cfg</code> ; adapt to marvosym’s changed encoding	193	<code>\SetExpansion</code> : new key: <code>inputenc</code>	113
		<code>\SetProtrusion</code> : (et al.) new key: <code>inputenc</code>	112

2006/09/09 **Version 1.9f**

Protrusion: fix: <code>euler-vm</code> did not load euler settings	190	<code>\MT@reset@context</code> : only reset context if it has actually been changed	103
<code>\MT@curr@list@name</code> : fix: <code>\MessageBreak</code> must not be expanded	98	<code>\MT@set@inh@list</code> : fix: forgotten comma in the features list	119
<code>\MT@gdef@n</code> : new macros: global variants	44	<code>\MT@set@named@keys</code> : new macro: set name first, simplify parsing of optional argument	114
<code>\MT@get@inh@list</code> : fix: input encoding must be set after the inheritance list has been parsed	91	<code>\SetProtrusion</code> : (et al.) set catcodes before parsing optional argument	112
<code>\MT@glet</code> : new macro	44		

2007/01/14 **Version 2.0**

General: compatibility with listings: set catcode of backslash to zero (reported by <i>Steven Bath</i>)	55	new: <code>smallcaps</code>	142
compatibility with soul: register <code>\textls</code> and <code>\lstyle</code>	55	<code>\DeclareMicrotypeBabelHook</code> : new command: interaction with <code>babel</code>	112
new option: <code>babel</code> , by default false (language-dependent setup suggested by <i>Ulrich Dirr</i>)	125	<code>\lstyle</code> : fix: font switches don’t pose a problem anymore	78
new option: <code>letterspace</code> , by default 100	127	fix: <code>letterspacing</code> commands may be nested	78
new package <code>letterspace</code> : a stripped-down version, containing the <code>letterspacing</code> commands only	1	new command: <code>letterspacing</code>	78
option ‘ <code>babel</code> ’: fix: switch off French <code>babel</code> ’s short-hands properly (reported by <i>Daniel Flipo</i>)	141	totally redone, using the new <code>\letterspacefont</code>	78
option ‘ <code>babel</code> ’: switch off Turkish <code>babel</code> ’s short-hands	141	<code>\MT@declare@sets</code> : fix: empty size list when redefining set	105
option ‘ <code>unit</code> ’, <code>\SetProtrusion</code> : deprecate value ‘ <code>relative</code> ’ completely	117	<code>\MT@is@symbol</code> : made even more robust	96
Documentation: add hint about how to increase <code>font_max</code> and <code>font_mem_size</code>	28	<code>\MT@load@inputenc</code> : sanitise catcodes before loading input encoding (problem with listings)	68
add hint about warning when tracking and expansion is applied to a font	28	<code>\MT@pdftex@no</code> : case 6: pdfTeX 1.40	40
add remark about ‘ <code>draft</code> ’ option disabling microtype (noted by <i>Michalis Miatidis</i>)	9	<code>\MT@setup@noligatures</code> : maybe disable <code>\MT@noligatures</code> after the preamble	140
qualify hint about web documents with regard to older pdfTeX versions	25	<code>\MT@split@name</code> : adjust to possible <code>letterspacing</code>	59
qualify hints about expansion error messages with regard to older pdfTeX versions	27	<code>\SetExtraKerning</code> : new command: additional kerning	114
Font sets: new: <code>footnotesize</code> and <code>scriptsize</code>	142	<code>\SetExtraSpacing</code> : new command: adjustment of interword spacing	113
		<code>\SetTracking</code> : new command: tracking	113
		<code>\textls</code> : new command: <code>letterspacing</code>	83
		starred version: remove spaces around text	83
		<code>\tracingmicrotypeinpdf</code> : new debug method: mark all fonts with PDF annotations	37

2007/01/21 **Version 2.1**

General: compatibility with pinyin: disable microtype in <code>\py@macron</code> (reported by <i>Sven Naumann</i>)	56	<code>\MT@get@ls@basefont</code> : redone: use <code>\pdfmatch</code> to make it bullet-proof	79
fix: letterspace package forgot to load <code>keyval</code>	41	<code>\MT@orig@pickupfont</code> : compatibility with CJK: also check for its definition	100
<code>\slig</code> : new command: protect ligatures in letter-spaced text	79	<code>\textls</code> : fix: use <code>\hmode@bgroup</code>	83

2007/07/14 **Version 2.2**

General: disable microtype if wordcount is loaded (reported by <i>Ross Hetherington</i>)	52	<code>\MT@is@composite</code> : more robust: expand exactly once	97
new option: <code>copyfonts</code>	126	<code>\MT@is@symbol</code> : expand once more (for frenchpro)	96
simplify key declarations	116	<code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code>	75
use catcode trickery for e-TeX test	38	<code>\MT@lua</code> : (basic) support for LuaTeX	41
Documentation: add hint about error message with pdfTeX 1.40	27	<code>\MT@pdftex@no</code> : case 7: pdfTeX 1.40.4	40
add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i>)	25	<code>\MT@preset@aux@space</code> : generalised	68
add overview	5	<code>\MT@set@all@pr</code> : (et al.) allow empty values	63
logo transparency and amusement	1	<code>\MT@set@inputenc@</code> : only load inputenc files if necessary	67
Font aliases: declare <code>chr</code> (<code>chmath</code>) as an alias of <code>bch</code> (reported by <i>Geoff Vallis</i>)	144	<code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in pdfTeX 1.40.4)	76
declare <code>fp9x</code> , <code>fp9j</code> (FPL Neu) as aliases of <code>ppl</code> [<code>xj</code>]	144	possibility to customise interword spacing	76
Font sets: default set for tracking: <code>smallcaps</code>	143	<code>\MT@setup@expansion</code> : warning if stretch or shrink aren't multiples of <code>step</code>	136
Inheritance: remove <code>'-'</code> → <code>'127'</code>	147	<code>\MT@setupfont</code> : don't call <code>\@enc@update</code> anymore	57
Protrusion: settings for Bitstream Letter Gothic	154	only add features that are available with the respective pdfTeX	57
Spacing: add sample	194	<code>\MT@setupfont@hook</code> : restore percent character if Galician <code>babel</code> is loaded	54
Tracking: add ligatures that are to be disabled	150	<code>\MT@the@pr@code@tr</code> : adjust protrusion of letterspaced fonts	63
<code>\DeclareMicrotypeVariants</code> : new command	110	<code>\MT@tracking</code> : remember fonts that shouldn't be letterspaced	74
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only	111	<code>\MT@tracking@</code> : fix: tracking couldn't be re-enabled	74
<code>\slig</code> : always defined	79	<code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode	139
<code>\MT@checklist@font</code> : fix: construct font name from characteristics	62	<code>\MT@with@babel@and@T</code> : also inspect class options	52
<code>\MT@copy@font</code> : optionally work on copies of fonts	57	<code>\pickup@font</code> : letterspace: setup inside group	100
<code>\MT@get@basefamily</code> : redone, working on font names and suffixes of arbitrary length	88	<code>\SetTracking</code> : new key <code>'no ligatures'</code> to disable ligatures of letterspaced fonts	113
<code>\MT@get@charwd</code> : subtract letterspacing amount from width	65	new keys <code>'spacing'</code> and <code>'outer spacing'</code> to adjust interword spacing (suggested by <i>Steven E. Harris</i>)	113
<code>\MT@get@ls@basefont</code> : fix again: remember base font in a macro	79	third argument may be empty	113
<code>\MT@ifdimen</code> : employ LuaTeX features if available	46	<code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code>	103
<code>\MT@ifint</code> : employ LuaTeX features if available	45		
<code>\MT@ifstreq</code> : employ LuaTeX features if available	47		
fix: \LaTeX version shouldn't use <code>\x</code> and <code>\y</code> (found by <i>Wiebke Petersen</i>)	47		

2007/12/23 **Version 2.3**

General: disable <code>\microtypecontext</code> in <code>hyperref's</code> <code>\pdfstringdef</code>	55	<code>\microtypecontext</code> : made robust (reported by <i>Stephan Hennig</i>)	103
fix: really switch off Turkish shorthands	141	<code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code>	88
new value for verbose option: <code>silent</code> (suggested by <i>Karl Berry</i>)	127	<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	108
turned some warnings into errors	127	<code>\MT@in@clist</code> : don't use <code>\x</code> (reported by <i>Peter Meier</i>)	48
Documentation: add kerning sample	18	<code>\MT@is@active</code> : support for extended Unicode (inputenc/utf8x resp. ucs) – experimental	95
add letterspacing illustration	17	<code>\MT@noligatures</code> : fix: set evaluation didn't work (bug introduced in v2.2)	85
<code>\do@subst@correction</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i>)	101	<code>\MT@plain</code> : letterspace: support for <code>eplain/miniltx</code>	38
<code>\slig</code> : redone: extract outer kerns from current letterspacing amount	79	<code>\MT@set@curr@os</code> : adjusting spaces made more reliable	76

<code>\MT@set@tr@codes</code> : also adjust tracking if protrusion is not enabled, and even for <code>letterspace</code> (reported by <i>Stephan Hennig</i>)	76	<code>\MT@SetTracking</code> : sanity check for value	113
possibility to customise outer kerning (suggested by <i>Stephan Hennig</i>)	76	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled	138
		<code>\MT@tr@outer@l</code> : only change pre outer space if it contains <code>shrink</code>	81
2008/02/29 Version 2.3a			
General: fix test for <code>soul</code> under plain \TeX	56	too old for extensions	139
Documentation: add hint about <code>babel</code> having to be loaded first	26	<code>\MT@fix@catcode</code> : fix catcodes earlier, and also for the <code>letterspace</code> package	35
add table of available and enabled features	7	<code>\MT@getkey</code> : fix: <code>key=val</code> in class options list	133
mention <code>soulutf8</code>	30	<code>\MT@set@codes</code> : generalised	63
Protrusion: adjust LMR quotation marks again	159	<code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded	54
<code>\MT@error@doesnt@work</code> : error messages if <code>pdfTeX</code> is			
2008/06/04 Version 2.3b			
<code>\MT@exp@gcs</code> : new macro: reduce save stack size	44	also check for its definition	100
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i>)	58	<code>\MT@requires@latex</code> : new macro	38
<code>\MT@get@size</code> : grouping	107	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i>)	76
<code>\MT@noligatures@</code> : fix: warning messages for unknown slots	85	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode	81
<code>\MT@orig@pickupfont</code> : compatibility with <code>CJKutf8</code> :		make <code>\spaceskip-aware</code> (<code>ragged2e</code>)	81
		<code>\MT@tr@outer@r@</code> : additional test for horizontal mode	82
2008/11/11 Version 2.3c			
General: <code>LuaTeX</code> supported by default	40	coding (reported by <i>Vasile Gaburici</i>)	148
Documentation: add hint about spacing being experimental	26	<code>\MT@detokenize@c</code> : fix: remove last space only (reported by <i>Ulrich Durr</i>)	45
add hint about partial incompatibility with <code>CJK</code>	27	<code>\MT@tr@outer@r@</code> : additional test for horizontal mode (reported by <i>Sveinung Heggen</i>)	82
Inheritance: add <code>\textcommabelow[STst]</code> to <code>QX</code> en-			
2009/03/27 Version 2.3d			
General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i>)	56	(reported by <i>Ulrich Durr</i>)	79
move setup to the very end (for <i>Colin Rourke</i>)	141	<code>\MT@setup@expansion</code> : default step: 1 for <code>pdfTeX</code> versions ≥ 1.40	135
<code>\ifMT@inannot</code> : use <code>pdftexcmds</code> for debugging	37	<code>\MT@tr@outer@r@</code> : don't use <code>\x</code> (reported by <i>Ulrich Durr</i>)	82
<code>\lststyle</code> : disable for <code>LuaTeX</code>	78	fix: don't adjust in math mode (reported by <i>Christoph Bier</i>)	82
make invalid in math mode	78	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i>)	82
<code>\microtypesetup</code> : select font after setup	130	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning	84
<code>\MT@check@active@set</code> : warning for missing default sets	129	<code>\textls</code> : make math mode aware	83
<code>\MT@lua</code> : update for <code>LuaTeX 0.36</code>	41		
<code>\MT@set@tr@codes</code> : allow zero tracking	75		
<code>\MT@set@tr@zero</code> : fix: allow switching off tracking			
2009/11/09 Version 2.3e			
Documentation: suggest to patch <code>\@verbatim</code> instead of <code>\verbatim</code>	26	<code>\MT@fix@fontdimen@six</code> : fix: gobbling settings with tracking failed (reported by <i>Leo</i>)	59
Expansion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	151	<code>\MT@setup@</code> : make space-unaware (requested by <i>Marcin Borkowski</i>)	52
Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set	142	<code>\MT@tikz@setup</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i>)	53
add T2A encoding	142	<code>\MT@tr@outer@r@</code> : fix: set current kerning and spacing again (found by <i>Lars Rönnbäck</i>)	82
Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	159		
Spacing: settings for T2A encoding (contributed by			

2010/01/10 **Version 2.4**

- General: new file `microtype.lua` containing the lua functions (contributed by *Élie Roux*) 43
- Protrusion: settings for T2A encoded Minion (contributed by *Karl Karlsson*) 159

2013/03/13 **Version 2.5**

- General: allow contexts for LuaTeX 116
- disable ‘DVIOoutput’ option for XeTeX 126
- fix: check whether ‘*(file)/(line)*’ list name already exists (reported by *Till A. Heilmann*) 116
- letterspacing with LuaTeX 0.62 74
- new files: `microtype-pdftex.def`, `microtype-xetex.def`, `microtype-luatex.def`, containing engine-specific definitions 38
- protrusion with XeTeX 40
- restore `\space` inside listings (reported by *Rolf Dieterich*) 55
- Documentation: add hint about LuaTeX compatibility 27
- add hint about spacing and `ragged2e` 26
- add hint about dtx source code 28
- include `microtype-logo.dtx` and `microtype-lssample.dtx` 218
- Font aliases: declare `lmsy` and `lmm` as aliases of `cmsy` resp. `cmm` (reported by *Jonas Hogstrom*) 143
- declare `zgmX` etc. (`garamondX`) as aliases of `ugm` 145
- declare Latin Modern Roman (OpenType version) as alias of `lmr` when `fontspec` is loaded 143
- declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version) 144
- Font sets: add EU1 and EU2 encodings 142
- Inheritance: add rudimentary list for EU1 and EU2 149
- Protrusion: add default lists for EU1 and EU2 157
- improvements to Computer Modern Roman italics (contributed by *Hendrik Vogt*) 162
- Tracking: add EU2 encoding to default list 150
- `\DeclareCharacterInheritance`: allow more than one encoding 119
- `\DeclareMicrotypeAlias`: ignore spaces 110
- `\ifMT@nofamily`: info if settings are not family-specific (suggested by *Hàn Thê Thành*) 62
- `\LoadMicrotypeFile`: remove all spaces in font name 111
- `\lsstyle`: fix: ensure to set up math fonts (reported by *RazorXsr*) 78
- `\MT@define@code@key@family`: compatibility with `fontspec`: remove its internal counter (reported by *Till A. Heilmann*) 114
- `\MT@define@code@key@font`: scrub `fontspec` feature count (found by *Meho R*) 115
- `\MT@do@font`: adapt for LuaTeX 50
- adapt for XeTeX 50
- `\MT@get@slot@:` adapt for LuaTeX (requested by *Georg Duffner*) 92
- adapt for XeTeX 92
- `\MT@if@outer@next`: fix: conflict with `amsmath` (reported by *Scott Pakin*) 82
- `\MT@info@missing@char`: fix error message for XeTeX (reported by *Juan Acevedo*) 65
- `\MT@is@charx`: compatibility with unicode 96
- `\MT@ledmac@setup`: fix to work with XeTeX (reported by *Maieul Rouquette*) 53
- `\MT@l@s@set@l@s`: allow formulas in optional argument to `\textls` (fix by *Heiko Oberdiek*) 84
- `\MT@microtypecontext`: fix: ensure to set up math fonts (reported by *RazorXsr*) 103
- `\MT@register@subst@font`: only register substituted font if it isn’t registered already (reported by *George Gratzler* and *Josep Maria Font*) 102
- `\MT@register@subst@font@cx`: only register if it isn’t registered already 102
- `\MT@scrubfeatures`: compatibility with `fontspec`: remove its internal counter 60
- `\MT@set@all@pr`: fix: remove space (found by *Meho R*) 63
- `\MT@set@pr@codes`: make info about generic settings encoding-specific (reported by *Sebastian Schubert*) 62
- `\MT@setup@spacing`: warning with `ragged2e` (reported by *Steffen Hoffmann*) 138
- `\MT@setupfont`: select font with `fontspec` (found by *Georg Duffner*) 56
- `\MT@setupfont@hook`: restore `\%` and `\#` when `mathastext` is loaded (found by *Seamus Bradley*) 54

2013/05/23 **Version 2.5a**

- General: use `luatexbase` instead of `luatextra` (contributed by *Élie Roux*) 43
- Documentation: add notes on typesetting the documentation 28
- include OpenType configuration files 199
- `\MT@afteraftergroup`: fix: get outer kerning and spacing of nested `letterspacing right` 77
- `\MT@get@slot@:` adapt to `luaotfload v2.2` (contributed by *Élie Roux*) 93
- `\MT@led@unhbox@line`: simplified 53
- `\MT@ledmac@setup`: support for `eledmac` 53
- `\MT@l@s@outer@k`: add marker for tightly nested letterspacing 85
- `\MT@set@tr@codes`: fix: load font for `fontspec` 76
- `\MT@xspace`: fix outer spacing problem with `xspace` (reported by *Dave*) 83

2016/05/01 **Version 2.6**

- General: load `luaotfload` with LuaTeX 43
- redefine `\MT@setupfont@hook` globally for problem with `tikzposter` (reported by *Sam Mason*) 54
- Documentation: add hint about partial incompatibility with `xeCJK` and `luatexja` 27
- missing characters printed with `Charis SIL` 199

suggest to use etoolbox to patch <code>\verbatim</code>	26	<code>\MT@is@xchar</code> : update for fontspec's TU encoding	96
Font sets: add TU encoding (notified by <i>Will Robertson</i>)	142	<code>\MT@ledmac@setup</code> : support for <code>reledmac</code>	53
add <code>si</code> and <code>scit</code> to <code>smallcaps</code> set (reported by <i>uli</i>)	142	<code>\MT@luatex@no</code> : update for LuaTeX 0.85 (renamed primitives)	41
new: <code>allmath-nott</code> and <code>alltext-nott</code> (requested by <i>Karl Berry</i>)	142	<code>\MT@noligatures@</code> : use <code>luaotfload</code> function to keep/inhibit ligatures	86
Inheritance: add TU encoding	149	<code>\MT@orig@pickupfont</code> : (in)compatibility with <code>luatexja</code> : disable unknown slots warnings (reported by <i>Max</i>)	99
Protrusion: add TU encoding to lists	157	(in)compatibility with <code>xeCJK</code> : disable unknown slots warnings (reported by <i>HcN</i>)	99
Tracking: add TU encoding to default list	150	compatibility with <code>xeCJK</code> : pretend that <code>CJK</code> wasn't loaded	100
<code>\DeclareMicrotypeSet</code> : ignore spaces	104	<code>\MT@set@tr@codes</code> : use <code>luaotfload</code> 's <code>kernfactor</code> feature if available	75
<code>\DeclareMicrotypeSetDefault</code> : ignore spaces	109	<code>\MT@xspace</code> : fix outer spacing problem with (not only) algorithm (reported by <i>Henning</i> and <i>Ronnie Marksch</i>)	83
<code>\DeclareMicrotypeVariants</code> : ignore spaces	110	<code>\UseMicrotypeSet</code> : ignore spaces	109
<code>\lststyle</code> : fix: ensure to set up math fonts (reported by <i>kleenstar</i>)	78		
<code>\microtypecontext</code> : allow activate shortcut (reported by <i>Karl Berry</i>)	103		
<code>\MT@declare@sets</code> : fix: undefine lists for redefining	105		
<code>\MT@do@font</code> : speed up for LuaTeX	50		
<code>\MT@engine</code> : fix test with LuaTeX 0.85	39		
<code>\MT@get@slot@</code> : fix: could fail with X _Y TeX (reported by <i>Christopher Schramm</i>)	92		
2016/05/14	Version 2.6a		
General: fixes for <code>letterspace</code> package with LuaTeX	49	<code>Vofß</code>	50
<code>\MT@do@font</code> : fix lua function (reported by <i>Herbert</i>)		<code>\MT@ls@fontspec@font</code> : fix for value of ± 1000	77
2017/07/07	Version 2.7		
General: drop <code>luatexbase</code> with recent L ^A TeX	43	<code>\MT@check@range@</code> : don't warn for override if conflicting list is loaded	123
warning with <code>minimal</code> class	52	<code>\MT@is@composite</code> : compatibility with L ^A TeX 2017/01/01 (<code>\DeclareUnicodeComposite</code>) (reported by <i>Ulrike Fischer</i> and <i>'jcr'</i>)	97
Documentation: mention that additional kerning does not work in math mode (discovered by <i>'Daniel'</i>)	18	<code>\MT@ls@fontspec@font</code> : fix for 'file: {font}' spec (reported by <i>Reinhard Kotucha</i>)	77
Font aliases: declare aliases for <code>newpx</code>	144	<code>\MT@permute@@@@@</code> : don't warn for override if conflicting list is loaded	122
declare aliases for <code>newtx</code>	144	<code>\MT@reset@ef@codes</code> : only reset <code>\efcodes</code> for older LuaTeX versions	70
declare aliases for <code>tempora</code>	144	<code>\MT@setup@expansion</code> : don't disable automatic expansion for DVI output with LuaTeX	135
declare aliases for <code>XCharter</code>	144	<code>\MT@tikz@setup</code> : compatibility with <code>tikz</code> (again)	53
declare Latin Modern Roman as alias of <code>lmr</code> with new L ^A TeX format (reported by <i>Ulrike Fischer</i>)	143	<code>\MT@warn@tracking@DVI</code> : don't warn for letterspacing in DVI mode with LuaTeX	139
Protrusion: automatically choose correct names for Charis SIL small caps (reported by <i>'ltcomdata'</i>)	211		
<code>\lststyle</code> : fix: prevent infinite loop with <code>psnffs</code> and <code>exscale</code> packages (reported by <i>user11126</i> , solution by <i>Ulrike Fischer</i>)	79		
2018/01/14	Version 2.7a		
General: disallow non-automatic expansion with LuaTeX	118	<code>\MT@get@highlevel</code> : test whether <code>\...default</code> is defined	106
<code>\MT@auto</code> : remove 'autoexpand' for LuaTeX 1.0.6 (reported by <i>Ulrike Fischer</i>)	135	<code>\MT@get@slot</code> : expand active characters earlier	91
with LuaTeX, non-automatic font expansion is no longer possible (as confirmed by <i>Hans Hagen</i>)	135	<code>\MT@info@nottracking@</code> : defer 'No tracking' message	60
		<code>\MT@is@active</code> : compatibility with <code>newunicodechar</code> (reported by <i>Nils Anders Danielsson</i>)	95
2019/02/28	Version 2.7b		
General: update lua function <code>microtype.info</code> after changes in <code>luaotfload</code> (reported by <i>Moritz Wemheuer</i> and <i>Ulrike Fischer</i>)	43	(reported by <i>Franz Wexler</i>)	148
Documentation: update hint about non-7-bit characters (notified by <i>Frank Mittelbach</i>)	26	<code>\MT@info@missing@char</code> : fix message for glyphs specified as names in X _Y TeX (reported by <i>Paolo Ney</i>)	65
Inheritance: add <code>textquotedblleft</code> ligature to OT4		<code>\MT@setupfont</code> : always select current font with X _Y TeX and LuaTeX (reported by <i>Paolo Ney</i> , solution by <i>Ulrike Fischer</i>)	56

2019/10/10	Version 2.7c		
	General: turn warning into info when overwriting the keepligature function (reported by <i>Andy N</i>) ..	86	<code>\MT@is@symbol</code> : take care of <code>\remove@tlig</code>
	<code>\MT@is@active</code> : compatibility with L ^A T _E X 2019/10/01	95	<code>\showhyphens</code> : compatibility with L ^A T _E X 2019/10/01 (reported by <i>Phelype Oleinik</i> and <i>Falk Hanisch</i>)
2019/11/18	Version 2.7d		
	<code>\MT@copy@font@</code> : in LuaT _E X, don't use the <code>\copyfont</code> primitive, but load the font anew (reported by <i>Paolo Polesana</i> and <i>Oliver Kopp</i>)	58	from list (reported by <i>Markus Kohm</i>)
	<code>\MT@register@subst@font@</code> : remove substitute font		<code>\MT@register@subst@font@cx</code> : remove substitute font from lists
2020/12/07	Version 2.8		
	General: letterspace works with e-T _E X only	38	<code>\lsstyle</code> : fix: enforce math setup, again
	compatibility with soul: patch for font change (reported by <i>Md Ayquassar</i>)	56	<code>\microtypecontext</code> : fix activate shortcut
	fix for luatexbase	43	ignore spaces
	Documentation: declare DVIoutput option deprec- ated	9	<code>\MT@do@font@</code> : fix for X _Y T _E X
	squash fake news about automatic font expansion with dvilualatex	8	simplify lua function
	Font aliases: declare aliases for step and domitian (notified by <i>Daniel Benjamin Miller</i>)	144	<code>\MT@fix@fontdimen@six</code> : try to fix zero <code>\fontdimen 6</code>
	declare aliases for stix and stix2 fonts	144	<code>\MT@if@luaotf@font@</code> : use lua function
	declare New Computer Modern as an alias of Latin Modern Roman	144	<code>\MT@ifstreq</code> : use X _Y T _E X's <code>\stringcmp</code>
	Font sets: default set for expansion: alltext-nott	143	<code>\MT@setup@expansion</code> : warning when expanding in DVI mode with LuaT _E X (reported by <i>d909</i>) ..
	default set for spacing: alltext-nott	143	<code>\MT@tr@set@space@</code> : simplified
			<code>\MT@tr@unit@</code> : fix: allow unit regardless whether letterspacing is set
			<code>\textmicrotypecontext</code> : ignore spaces

D Index

Numbers in upright shape refer to the page where the corresponding entry is described (bold face) resp. occurs. Numbers in italics refer to the code line where the corresponding entry is defined (underlined) resp. used.

Options	DVIoutput	9	letterspace	8
	activate	6	protrusion	6
	auto	7	selected	8
	babel	9	shrink	8
	config	9	spacing	6
	draft	9	step	8
	expansion	6	stretch	8
	factor	7	tracking	6
	final	9	unit	7
	kerning	6	verbose	9
Commands	<code>\DeclareCharacterInheritance</code>	20	<code>\SetExtraSpacing</code>	19
	<code>\DeclareMicrotypeAlias</code>	21	<code>\SetProtrusion</code>	13
	<code>\DeclareMicrotypeBabelHook</code>	23	<code>\SetTracking</code>	15
	<code>\DeclareMicrotypeSet*</code>	10	<code>\UseMicrotypeSet</code>	12
	<code>\DeclareMicrotypeSet</code>	10	<code>\lslig</code>	24
	<code>\DeclareMicrotypeSetDefault</code>	12	<code>\lsstyle</code>	23
	<code>\DeclareMicrotypeVariants*</code>	20	<code>\microtypecontext</code>	22
	<code>\DeclareMicrotypeVariants</code>	20	<code>\microtypesetup</code>	9
	<code>\DisableLigatures</code>	24	<code>\textls*</code>	23
	<code>\LoadMicrotypeFile</code>	22	<code>\textls</code>	23
	<code>\SetExpansion</code>	14	<code>\textmicrotypecontext</code>	22
	<code>\SetExtraKerning</code>	18		

- A** aoposter (package) 107
 activate (option) 6, 125, 232
 \add@accent 2941
 \adjustspacing 196, 4322
 ae (package) 21, 143
- B** babel (option) 9, 23, 27, 31, 125, 232
 babel (package) 2, 3, 5, 17, 23, 24, 27,
- C** chemsym (package) 229
 chmath (package) 21, 144, 233
 CJK (package) 27, 56, 100, 233, 234, 236
 CJKutf8 (package) 100, 234
 cm-super (package) 8
 color (package) 133
 combine (package) 141, 232
- D** \DeclareCharacterInheritance 20, 41, 3767
 \DeclareMicrotypeAlias 21, 43, 71, 3359
 \DeclareMicrotypeBabelHook 23, 45, 64, 3425
 \DeclareMicrotypeSet 10, 32, 57, 59, 72, 3073
 \DeclareMicrotypeSet* 10, 3073
 \DeclareMicrotypeSetDefault 12, 34, 61, 3314
 \DeclareMicrotypeVariants 20, 42, 58, 63, 3344
 \DeclareMicrotypeVariants* 20
 defersetup (option) 54, 125, 126, 130, 231
- E** eco (package) 21, 143
 \efcode 1551, 1577, 1578, 1619, 1621
 eledmac (package) 53, 235
 eplain (package) 24, 31, 34, 38, 52, 133, 233
 e-TeX (engine) 33, 38, 40,
 44, 45, 50, 51, 64, 82, 218, 229–231, 233, 237
 etoolbox (package) 26, 236
 eucal (package) 190
 eufrak (package) 192
- F** \f@family 2892, 2893
 \f@size 1925, 2885, 2887, 2889, 2896, 2939
 factor (option) 7, 13, 32, 127, 229
 fancyvrb (package) 26
 final (option) 9, 33, 125, 127, 228
 fix-cm (package) 59
 \font 966, 1042, 1827, 1975
 \font@name 111, 1014, 1016, 1018, 1020,
 1021, 1031, 1032, 1817, 1825, 1829, 1833,
 1871, 1877, 1882, 1885, 1960, 1975, 1987,
 1989, 1994, 1996, 1999, 2002, 2005, 2040,
 2091, 2228, 2230, 2878, 2886, 2895, 2900,
 2920, 2922, 2924, 2940, 2954, 2956, 2976, 2981
- G** garamondx (package) 21, 145, 235
 german (package) 35, 228
 \glb@currsiz 1963, 3015
- H** hfoldsty (package) 21, 143
- I** IEEEtran (package) 99
 \iffontchar 727, 1325, 2814, 2826, 2827
 \ifMT@auto 301, 4526, 4587
 \ifMT@babel 301, 4673, 4801
 \ifMT@do 1081, 1148, 1785, 2275
 \ifMT@document 342, 3878
 \ifMT@draft 301, 4419
 \ifMT@expansion 301, 4475, 4489, 4584, 4630
 \ifMT@fontspec 846, 1061, 3144, 3549, 3575
 \ifMT@if@ 300,
 863, 870, 893, 925, 2972, 3985, 4821, 4824
 \ifMT@inannot 99
- algorithm (package) 236
 amsmath (package) 235
 amssymb (package) 185
 article (package) 22
 auto (option) 7, 8, 125
 54, 88, 112, 140, 141, 145, 229, 230, 232–234
- config (option) 9, 20, 32, 128, 230
 contour (package) 133
 copyfonts (option) 58, 125, 126, 233
 crop (package) 133
 csquotes (package) 29, 54, 157, 230
 \curr@fontshape 2885, 2887, 2889, 2896, 2939
- \define@newfont 2878, 2886, 2890, 2900
 \DisableLigatures 24, 40, 62, 3398
 \do@subst@correction 2938
 docstrip (package) 34
 domitian (package) 21, 144, 237
 draft (option) 9, 125, 127, 232
 dsfont (package) 59
 DVIOutput (option) 9, 125, 126, 235, 237
- euler (package) 98, 189, 232
 eulervm (package) 21, 145, 190, 232
 euroitc (package) 193
 europms (package) 193
 eurosans (package) 193
 \expandglyphsinfont 1533
 expansion (option) 6, 11, 124, 125, 134
 exscale (package) 236
- \fontcharwd 1279, 1283, 1289
 \fontdimen 1046, 1047,
 1051, 1052, 1055, 1314, 1319, 1419, 1422,
 1846, 1847, 1877, 1987, 2033, 2034, 2041,
 2042, 2049, 2051, 2057, 2072, 2091, 2241, 2244
- fontinst (package) 197, 218, 220
 fontinstallationguide (package) 175
 \fontname 1825, 1829
 fontspec (package) 6, 25, 27, 30,
 43, 59, 60, 77, 96, 106, 114, 143, 149, 235, 236
 fourier (package) 59
 french (package) 229
 frenchpro (package) 96, 229, 233
- \glb@settings 1964
 graphics (package) 133
- hyperref (package) 54, 55, 99, 133, 230, 233
- \ifMT@inlist@ 620, 665, 773, 803, 986, 1115,
 1132, 1141, 1159, 1177, 1783, 2379, 2390,
 2439, 2488, 2955, 2966, 2978, 3043, 3055, 3381
- \ifMT@kerning 301, 4687, 4814
 \ifMT@nofamily 1187, 1194
 \ifMT@noligatures 301, 4767
 \ifMT@nonselected 1513, 2501, 2509
 \ifMT@norest 2570, 2587, 2600, 2614, 2675
 \ifMT@opt@auto 3996, 4530, 4562
 \ifMT@opt@DVI 3996, 4444
 \ifMT@opt@expansion 3996, 4473, 4631
 \ifMT@protrusion 301, 810, 833, 4456, 4646

- `\ifMT@selected` [301](#), [4570](#), [4590](#)
`\ifMT@spacing` [301](#), [4655](#), [4672](#)
`\ifMT@tracking` [301](#), [4642](#), [4702](#)
`\ifMT@xunicode` [846](#), [2765](#), [2818](#)
`ifpdf` (package) [100](#), [133](#)
- J**
- `jurabib` (package) [99](#)
- K**
- `kerneling` (option) [6](#), [11](#), [31](#), [125](#)
`keyval` (package) [41](#), [63](#), [146](#), [233](#)
`\knacode` [1734](#), [1735](#), [1745](#), [1748](#), [1754](#)
- L**
- `ledmac` (package) [29](#), [32](#), [52](#), [99](#), [230](#)
`ledpar` (package) [29](#)
`\leftmarginkern` [816](#)
`letterspace` (option) [8](#), [17](#), [23](#), [24](#), [31](#), [45](#), [75](#), [127](#), [232](#)
`letterspace` (package) [1](#), [24](#), [31](#), [32](#),
[34](#), [36](#), [38](#), [53](#), [76](#), [77](#), [79](#), [232–234](#), [236](#), [237](#)
`\letterspacefont` [1833](#)
`lineno` (package) [52](#)
`listings` (package) [26](#), [55](#), [68](#), [88](#), [231](#), [232](#), [235](#)
`lmodern` (package) [8](#), [159](#)
`\LoadMicrotypeFile` [22](#), [44](#), [74](#), [3377](#), [4424](#)
`\lpcode` [1216](#), [1252](#), [1253](#), [1457](#), [1460](#), [1860](#)
- M**
- `marvosym` (package) [21](#), [32](#), [34](#), [150](#), [193](#), [231](#), [232](#)
`mathastext` (package) [54](#), [235](#)
`mathdesign` (package) [21](#), [145](#), [231](#)
`memoir` (package) [22](#), [99](#)
`\Microtype@Hook` [129](#), [4253](#)
`\microtypecontext`
. [22](#), [47](#), [914](#), [3004](#), [3023](#), [4426](#), [4784](#), [4787](#)
`microtypecontext` (environment) [22](#)
`\microtypesetup` [9](#), [46](#), [4257](#), [4425](#)
`MiKTeX` (distribution) [28](#)
`miniltx` (package) [24](#), [31](#), [34](#), [38](#), [52](#), [233](#)
`minimal` (package) [52](#), [56](#), [236](#)
`MinionPro` (package) [106](#)
`MT@efont` [402](#), [989](#), [1012](#), [1051](#), [1053](#), [1315](#),
[1329](#), [1797–1799](#), [1801](#), [1804](#), [2442](#), [2514](#), [2925](#)
`\MT@abbr@ex` [749](#)
`\MT@abbr@ex@c` [749](#)
`\MT@abbr@ex@inh` [749](#)
`\MT@abbr@kn` [749](#)
`\MT@abbr@kn@c` [749](#)
`\MT@abbr@kn@inh` [749](#)
`\MT@abbr@nl` [749](#)
`\MT@abbr@pr` [749](#)
`\MT@abbr@pr@c` [749](#)
`\MT@abbr@pr@inh` [749](#)
`\MT@abbr@sp` [749](#)
`\MT@abbr@sp@c` [749](#)
`\MT@abbr@sp@inh` [749](#)
`\MT@abbr@tr` [749](#)
`\MT@abbr@tr@c` [749](#)
`\MT@active@features`
. [1023](#), [2960](#), [2963](#), [2975](#), [2988](#), [3032](#), [3042](#),
[3406](#), [4457](#), [4585](#), [4643](#), [4656](#), [4688](#), [4776](#), [4777](#)
`\MT@addto@annot` [99](#)
`\MT@addto@setup` [794](#),
[795](#), [874](#), [1432](#), [2869](#), [3005](#), [3006](#), [3636](#),
[3654](#), [4110](#), [4258](#), [4418](#), [4733](#), [4775](#), [4800](#), [4833](#)
`\MT@adjustspacing` [4318](#), [4319](#), [4586](#)
`\MT@afteraftergroup` [1889](#), [1897](#), [1906](#)
`\MT@auto` [1523](#), [4525](#), [4540](#), [4550](#), [4553](#), [4557](#)
`\MT@auto@` [1523](#), [1546](#), [1591](#)
`\MT@autofalse` [303](#), [4538](#), [4549](#)
- `\iftracingmicrotypeinpdfall` [105](#)
`\ignorereligaturesinfont` [2279](#)
`inputenc` (package) [14](#),
[26](#), [32](#), [55](#), [67](#), [68](#), [91](#), [95](#), [98](#), [230](#), [232](#), [233](#)
- `\knbcode` [1727](#), [1728](#), [1744](#), [1747](#), [1753](#)
`\knbscode` [1649](#), [1650](#), [1673](#), [1677](#), [1683](#)
- `\lslig` [24](#), [54](#), [1980](#), [1983](#)
`\lssstyle` [23](#), [51](#), [916](#), [940](#),
[1961](#), [2167](#), [2173](#), [2178](#), [2181](#), [2191](#), [2195](#), [4427](#)
- `luaotfload` (package)
. [27](#), [30](#), [43](#), [49](#), [70](#), [75](#), [86](#), [93](#), [235](#), [236](#)
- `LuaTeX` (engine) [1](#), [4](#), [6–9](#), [12–16](#),
[20](#), [23–25](#), [27–31](#), [34](#), [37–41](#), [43](#), [45](#), [50](#), [53](#),
[56](#), [58](#), [70](#), [74](#), [76](#), [83](#), [85](#), [86](#), [92](#), [116](#), [126](#),
[132](#), [134](#), [135](#), [139](#), [143](#), [146](#), [149](#), [157](#), [233–237](#)
- `luatexbase` (package) [43](#), [235–237](#)
`luatexja` (package) [27](#), [99](#), [235](#), [236](#)
`luatextra` (package) [235](#)
- `\MT@autotruer` [303](#), [4212](#), [4215](#), [4565](#)
`\MT@babelfalse` [310](#)
`\MT@babeltrue` [310](#)
`\MT@begin@catcodes`
. [2380](#), [2381](#), [2421](#), [3004](#), [3008](#), [3021](#),
[3074](#), [3284](#), [3315](#), [3346](#), [3360](#), [3385](#), [3401](#),
[3435](#), [3451](#), [3473](#), [3494](#), [3508](#), [3773](#), [4228](#), [4229](#)
`\MT@cat` [1428](#), [2860](#), [2862](#)
`\MT@cfg@catcodes` [927](#), [1450](#), [2406](#), [2423](#)
`\MT@char` [1241](#), [1242](#),
[1252](#), [1253](#), [1258](#), [1259](#), [1262](#), [1264](#), [1279](#),
[1280](#), [1283](#), [1457–1461](#), [1577](#), [1578](#), [1580](#),
[1581](#), [1619–1621](#), [1649](#), [1650](#), [1656](#), [1657](#),
[1663](#), [1664](#), [1667](#), [1668](#), [1673–1678](#), [1727](#),
[1728](#), [1734](#), [1735](#), [1738](#), [1739](#), [1744–1748](#),
[2287](#), [2288](#), [2290](#), [2543](#), [2559](#), [2567](#), [2569](#),
[2579](#), [2581](#), [2589](#), [2590](#), [2595](#), [2599](#), [2600](#),
[2605](#), [2607](#), [2613](#), [2615](#), [2618](#), [2622](#), [2626](#),
[2630](#), [2746](#), [2749](#), [2751](#), [2753](#), [2809](#), [2814](#),
[2816](#), [2819](#), [2821](#), [2827](#), [3823–3825](#), [3830–3832](#)
`\MT@char@` [1285](#), [1286](#), [1289](#), [1324](#),
[1325](#), [2543](#), [2550](#), [2554](#), [2559](#), [2608](#), [2679](#),
[2681](#), [2687](#), [2688](#), [2690](#), [2703](#), [2704](#), [2707](#),
[2708](#), [2711](#), [2712](#), [2716](#), [2718](#), [2750](#), [2752](#),
[2772](#), [2775](#), [2779](#), [2785](#), [2792](#), [2794](#), [2817](#), [2834](#)
`\MT@charstring` [2556](#), [2749](#), [2757](#), [2784](#)
`\MT@charxstring` [2787](#), [2821](#)
`\MT@check@active@set`
. [4239](#), [4463](#), [4603](#), [4645](#), [4665](#), [4692](#)
`\MT@check@font` [985](#), [2951](#), [3034](#)
`\MT@check@font@cx` [2961](#), [3034](#)
`\MT@check@range` [3956](#), [3958](#)
`\MT@check@range@` [3958](#), [3959](#)
`\MT@check@rlist` [3905](#), [3949](#)
`\MT@check@rlist@` [3949](#), [3950](#)
`\MT@check@step` [4591](#)
`\MT@checklist@` [1089](#), [1108](#), [2272](#)
`\MT@checklist@family` [1126](#)
`\MT@checklist@font` [1170](#)
`\MT@checklist@size` [1154](#)
`\MT@checksetup` [4269](#), [4280](#), [4287](#), [4306](#), [4342](#)
`\MT@clear@options` [163](#), [293](#), [784](#), [4201](#), [4411](#)

- `\MT@clist@break` [592](#), [1121](#), [1149](#), [1165](#), [1179](#)
`\MT@clist@function` [592](#)
`\MT@cnt@encoding` [3844](#), [3852](#), [3853](#)
`\MT@cnt@family` [3850](#), [3859](#), [3860](#)
`\MT@cnt@series` [3857](#), [3866](#), [3867](#)
`\MT@cnt@shape` [3864](#), [3872](#), [3873](#)
`\MT@config@file` [4217](#), [4226](#), [4227](#), [4231](#), [4232](#), [4235](#)
`\MT@context` [2464](#), [2476](#), [2495](#), [2522](#)
`\MT@copy@font` [972](#),
[1008](#), [3632](#), [3638](#), [3650](#), [3657](#), [4119](#), [4121](#), [4447](#)
`\MT@copy@font@` . [1008](#), [3632](#), [3638](#), [3650](#), [3657](#), [4119](#)
`\MT@count` [508](#), [509](#),
[733](#), [1271](#), [1279](#), [1281](#), [1283](#), [1287](#), [1289](#),
[1292](#), [1298](#), [1303](#), [1304](#), [1308](#), [1319](#), [1351](#), [1413](#)
`\MT@curr@file` [2384](#), [2385](#), [2394](#), [2395](#),
[3387](#), [3388](#), [3606](#), [3608](#), [3611](#), [3795](#), [4231](#), [4835](#)
`\MT@curr@list@name` . . . [1317](#), [1354](#), [1444](#), [1587](#),
[2284](#), [2365](#), [2533](#), [2829](#), [2837](#), [2842](#), [2850](#), [2856](#)
`\MT@curr@ls` [1868](#), [1984](#), [1987](#)
`\MT@curr@set@name` [3098](#), [3100](#)–
[3102](#), [3104](#), [3107](#), [3108](#), [3113](#), [3118](#), [3122](#),
[3123](#), [3132](#), [3157](#), [3161](#), [3190](#), [3220](#), [3226](#),
[3232](#), [3236](#), [3237](#), [3488](#), [3564](#), [3585](#), [3600](#),
[3619](#), [3691](#), [3695](#), [3702](#), [3705](#), [3707](#), [3713](#),
[3716](#), [3719](#), [3731](#), [3745](#), [3754](#), [3762](#), [3796](#), [3798](#)
`\MT@declare@char@inh` [3778](#), [3783](#), [3793](#)
`\MT@declare@sets` [3081](#), [3086](#), [3099](#), [3408](#)
`\MT@DeclareMicrotypeAlias` [3361](#), [3363](#)
`\MT@DeclareMicrotypeSetDefault` [3316](#), [3318](#)
`\MT@DeclareSet` [3077](#), [3079](#), [3095](#)
`\MT@DeclareSetAndUseIt` [3076](#), [3094](#)
`\MT@DeclareVariants` [3348](#), [3349](#), [3351](#)
`\MT@def@bool@opt` [4073](#),
[4086](#), [4087](#), [4091](#), [4105](#), [4117](#), [4127](#), [4139](#)
`\MT@def@n` [389](#), [3069](#), [3070](#)
`\MT@default@ex@set` [3333](#)
`\MT@default@kn@set` [3333](#)
`\MT@default@pr@set` [3333](#)
`\MT@default@sp@set` [3333](#)
`\MT@default@tr@set` [3333](#)
`\MT@define@code@key`
. . . [3532](#), [3592](#), [3594](#), [3595](#), [3805](#), [3807](#), [3808](#)
`\MT@define@code@key@family` [3543](#), [3593](#), [3806](#)
`\MT@define@code@key@font` [3569](#), [3597](#), [3810](#)
`\MT@define@code@key@size` [3557](#), [3596](#), [3809](#)
`\MT@define@opt@key` [3598](#), [3621](#)–[3624](#), [3758](#)–[3760](#)
`\MT@define@optionX` [4257](#), [4315](#), [4318](#)
`\MT@define@optionX@` [4332](#), [4361](#), [4363](#), [4364](#)
`\MT@define@set@key@` [3111](#), [3276](#)–[3279](#)
`\MT@define@set@key@font` [3224](#), [3281](#)
`\MT@define@set@key@size` [3150](#), [3280](#)
`\MT@detokenize@c` [432](#), [2551](#), [2747](#)
`\MT@detokenize@n` [432](#), [2811](#)
`\MT@dimen@six`
. . . [1044](#), [1272](#), [1308](#), [1351](#), [2066](#), [2068](#), [2237](#)
`\MT@dinfo` [87](#)
`\MT@dinfo@list`
[1106](#), [1116](#), [1119](#), [1124](#), [1133](#), [1136](#), [1142](#),
[1144](#), [1152](#), [1160](#), [1163](#), [1168](#), [1178](#), [1181](#), [1185](#)
`\MT@dinfo@nl` [87](#)
`\MT@DisableLigatures` [3398](#)
`\MT@do@font` [700](#), [1218](#), [1551](#), [1686](#), [1755](#), [1860](#)
`\MT@document@false` [342](#)
`\MT@document@true` [342](#), [4779](#)
`\MT@dofalse` [1081](#), [1093](#), [1120](#), [1137](#), [1164](#), [1182](#)
`\MT@dofont@function` [704](#), [715](#)
`\MT@dotrue` [1081](#), [1084](#), [1117](#), [1134](#), [1143](#), [1161](#), [2267](#)
`\MT@draft@false` [306](#), [4141](#)
`\MT@draft@true` [306](#), [4143](#)
`\MT@edef@n` [391](#), [2500](#), [2526](#), [3059](#),
[3538](#), [3552](#), [3608](#), [3611](#), [3614](#), [3794](#), [3936](#), [4173](#)
`\MT@encoding` [1058](#), [1174](#), [1195](#), [1197](#), [1200](#),
[2459](#), [2471](#), [2551](#), [2747](#), [2810](#), [2841](#), [2848](#), [2856](#)
`\MT@end@catcodes` [2382](#), [2425](#), [3005](#), [3012](#),
[3022](#), [3092](#), [3300](#), [3331](#), [3357](#), [3375](#), [3393](#),
[3410](#), [3446](#), [3470](#), [3489](#), [3505](#), [3519](#), [3788](#), [4230](#)
`\MT@end@input` [167](#), [4196](#), [4205](#)
`\MT@engine` [175](#),
[274](#), [278](#), [284](#), [290](#), [2168](#), [2172](#), [4197](#), [4203](#)
`\MT@engine@tooold` [175](#), [275](#)
`\MT@error` [76](#), [776](#), [877](#),
[2168](#), [2179](#), [2357](#), [2370](#), [3188](#), [3309](#), [3338](#),
[3416](#), [3641](#), [3667](#), [3674](#), [4129](#), [4292](#), [4310](#),
[4350](#), [4531](#), [4543](#), [4563](#), [4632](#), [4703](#), [4716](#), [4796](#)
`\MT@error@doesnt@work` [4699](#)
`\MT@ex@c@name` [1505](#),
[1507](#), [1593](#), [1594](#), [1605](#), [1611](#), [1612](#), [1624](#), [3454](#)
`\MT@ex@context` [1012](#), [3026](#), [3068](#)
`\MT@ex@doc@contexts` [3068](#)
`\MT@ex@factor` [311](#), [1524](#), [1597](#)
`\MT@ex@factor@` . [1524](#), [1553](#), [1558](#), [1567](#), [1568](#), [1591](#)
`\MT@ex@inh@name` [1579](#)–[1581](#)
`\MT@ex@level` [311](#), [4586](#), [4588](#)
`\MT@ex@max` [323](#), [1570](#), [1571](#)
`\MT@ex@min` [323](#), [1573](#), [1574](#)
`\MT@ex@setname` [3302](#)
`\MT@ex@split@val` [1565](#)
`\MT@exp@cs` [386](#), [389](#), [392](#), [395](#), [400](#), [1024](#),
[1026](#), [1035](#), [1158](#), [1263](#), [1338](#), [1341](#), [1371](#),
[1375](#), [1385](#), [1388](#), [1391](#), [1581](#), [1668](#), [1739](#),
[1909](#)–[1911](#), [2487](#), [2979](#), [2989](#), [2990](#), [2994](#),
[3045](#), [3056](#), [3156](#), [3563](#), [3583](#), [3832](#), [3907](#), [3954](#)
`\MT@exp@gcs` [386](#), [390](#), [394](#), [396](#), [401](#)
`\MT@exp@one@n` [403](#), [437](#), [599](#), [976](#),
[978](#), [1036](#), [1112](#), [1175](#), [1782](#), [2389](#), [2951](#),
[2954](#), [2964](#), [2976](#), [2982](#), [3000](#), [3042](#), [3053](#),
[3068](#), [3086](#), [3275](#), [3294](#), [3325](#), [3380](#), [3783](#), [4784](#)
`\MT@exp@two@c` [405](#), [423](#), [428](#), [438](#), [634](#), [975](#),
[1016](#), [1018](#), [1020](#), [1021](#), [1030](#), [1828](#), [1960](#),
[1996](#), [1999](#), [2002](#), [2548](#), [2555](#), [2748](#), [2819](#), [2820](#)
`\MT@exp@two@n` [407](#), [1130](#), [1139](#), [3117](#), [3231](#)
`\MT@expand@font` [1509](#), [1527](#), [1531](#)
`\MT@expansion` [992](#), [1496](#), [4623](#)
`\MT@expansion@false` [302](#), [4481](#), [4486](#), [4580](#)
`\MT@expansion@true` [302](#), [4211](#)
`\MT@extra@context` [3068](#), [3438](#), [3454](#), [3477](#),
[3497](#), [3511](#), [3578](#), [3581](#), [3582](#), [3584](#), [3625](#),
[3633](#), [3639](#), [3651](#), [3656](#), [3769](#), [3904](#), [3908](#),
[3911](#), [3914](#), [3915](#), [3920](#), [3925](#), [3926](#), [3928](#), [3955](#)
`\MT@extra@input@enc` [3767](#), [3792](#), [3797](#), [3798](#)
`\MT@factor@default` [335](#), [4180](#), [4460](#)
`\MT@family` [976](#), [1058](#),
[1131](#), [1174](#), [1195](#), [1197](#), [1200](#), [2460](#), [3371](#), [3372](#)
`\MT@family@alias`
[977](#), [978](#), [1070](#), [1138](#), [1140](#), [2470](#), [2472](#), [3373](#)
`\MT@feat` [1081](#), [1244](#), [1333](#), [1335](#), [1337](#),
[1338](#), [1340](#), [1341](#), [1347](#), [1349](#), [1352](#), [1360](#)–
[1364](#), [1366](#), [1368](#)–[1372](#), [1375](#), [1376](#), [1381](#),
[1385](#), [1388](#), [1391](#), [1394](#)–[1396](#), [1405](#), [1406](#),
[1411](#), [1429](#), [1803](#), [1965](#), [2355](#), [2357](#), [2360](#),

- 2361, 2365, 2367, 2370, 2496, 2498, 2500,
 2504, 2508, 2512, 2513, 2516, 2524, 2526,
 2527, 2529, 2536, 2540, 2830, 2831, 2860, 2862
 \MT@features
 769, 3068, 3081, 3275, 3289, 3320, 3603, 3778
 \MT@features@long 769, 772, 777, 3039, 3790
 \MT@file@list 2376,
 2378, 2386, 2389, 2391, 2396, 2399, 3380, 3384
 \MT@find@file 976, 978, 2376
 \MT@fix@catcode 5
 \MT@fix@font@set 3140, 4431
 \MT@fix@fontdimen@six 1044, 1069
 \MT@font 402, 727, 729,
 969, 975, 1030, 1037, 1046–1049, 1052,
 1055, 1216, 1217, 1252, 1253, 1258, 1259,
 1279, 1280, 1283, 1286, 1289, 1314, 1319,
 1325, 1419, 1422, 1457, 1458, 1460, 1461,
 1536, 1540, 1546, 1551, 1577, 1578, 1619,
 1621, 1649, 1650, 1656, 1657, 1663, 1664,
 1673–1675, 1677, 1678, 1683–1685, 1727,
 1728, 1734, 1735, 1744, 1745, 1747, 1748,
 1753, 1754, 1782, 1786, 1866, 1867, 2276,
 2566, 2621, 2864, 2920–2922, 2924, 2939,
 2940, 2951, 2952, 2957, 2964, 2982, 2992, 3001
 \MT@font@copy . 1012, 1020, 1022, 1030–1032, 1042
 \MT@font@list
 .. 2864, 2951, 2952, 2954, 2956, 2957, 3033
 \MT@font@orig 1014
 \MT@font@sets 3121, 3140, 3235, 4431
 \MT@fontspecfalse 848
 \MT@fontspectrue 848, 849, 887
 \MT@gdef@n 389, 3429, 3445,
 3469, 3504, 3518, 3707, 3731, 3745, 3800, 4247
 \MT@get@axis 3249–3252, 3262
 \MT@get@basefamily 2388, 2426
 \MT@get@basefamily@ 2432, 2435
 \MT@get@char@unit 1243, 1383, 1409, 1501
 \MT@get@charwd 1277, 1308, 1386, 1409
 \MT@get@config 4217
 \MT@get@ex@opt 1500, 1518, 1591
 \MT@get@ex@opt@ 1599–1601, 1603, 1610
 \MT@get@font 3230, 3240
 \MT@get@font@ 3241, 3246, 3590
 \MT@get@font@and@size 3574, 3589
 \MT@get@font@dimen 1312, 1389
 \MT@get@highlevel ... 3116, 3126, 3264, 3537, 3548
 \MT@get@inh@list ... 1205, 1503, 1636, 1714, 2522
 \MT@get@listname 2441, 2498, 2524
 \MT@get@listname@ 2441
 \MT@get@ls@basefont 1821, 1986, 1993
 \MT@get@opt 1203, 1358, 1634, 1712
 \MT@get@range 3154, 3164, 3561
 \MT@get@size 3171, 3176, 3185, 3204, 3260
 \MT@get@slot 1240, 2286, 2543, 3822, 3829
 \MT@get@slot@ 2560, 2564
 \MT@get@space@unit
 .. 1383, 1485, 1647, 1654, 1661, 1725, 1732
 \MT@get@tr@opt 1808, 1928
 \MT@get@tr@opt@ 1948–1951, 1953
 \MT@get@unit 1391, 1399, 1937
 \MT@get@unit@ 1399
 \MT@get@key 4398, 4413
 \MT@glet . 384, 396, 888, 1016, 1910, 1996, 2027,
 2173, 2181, 2918, 3015, 3034–3037, 3373,
 3632, 3650, 3847, 4119, 4121, 4747, 4753, 4839
 \MT@glet@enc
 395, 1053, 1200, 1804, 2040, 2228, 2536,
 2940, 3033, 3104, 3107, 3113, 3226, 3462,
 3691, 3695, 3702, 3705, 3713, 3716, 3719, 3754
 \MT@glet@enn 400, 3052, 4244
 \MT@if@false 300,
 859, 866, 889, 906, 2969, 3953, 4815, 4822
 \MT@if@list@exists
 .. 1193, 1499, 1517, 1633, 1711, 1807, 2495
 \MT@if@luaotf@font
 .. 683, 703, 1823, 1858, 1867, 2289, 2302
 \MT@if@outer@next
 .. 2107, 2122, 2126, 2129, 2134, 2139–2142
 \MT@if@true 300,
 860–862, 867–869, 890–892, 919, 923, 924,
 2962, 3962, 3966, 3974, 3979, 4816–4820, 4823
 \MT@ifdefined@c@T 411, 863, 895,
 1261, 1297, 1579, 1666, 1737, 1852, 1942,
 2734, 3142, 3371, 3797, 3894, 4253, 4256, 4842
 \MT@ifdefined@c@TF
 .. 411, 811, 1811, 1836, 1839, 1850,
 1984, 2063, 2234, 2281, 2282, 2499, 2525, 3903
 \MT@ifdefined@n@T
 . 411, 800, 1109, 1127, 1155, 1171, 1262,
 1394, 1430, 1580, 1605, 1667, 1738, 1932,
 1940, 1954, 2478, 2859, 3101, 3366, 3457,
 3615, 3853, 3860, 3867, 3873, 3904, 3914, 4398
 \MT@ifdefined@n@TF 411, 1070,
 1087, 1110, 1128, 1156, 1172, 1195, 1360,
 1368, 1593, 1611, 1798, 1908, 2270, 2360,
 2466, 2551, 3129, 3303, 3306, 3334, 3606,
 3880, 3935, 3944, 4014, 4240, 4243, 4604, 4782
 \MT@ifdim .. 531, 654, 655, 659, 660, 3187, 3196,
 3960–3962, 3964, 3965, 3972–3974, 3977, 3978
 \MT@ifdimen 493, 3215, 3694, 3704, 3718, 4189
 \MT@ifempty 444, 1216, 1217, 1250, 1256,
 1474, 1475, 1646, 1653, 1660, 1683–1685,
 1697–1699, 1701–1703, 1724, 1731, 1753,
 1754, 1770, 1771, 2046, 2047, 2055, 2071,
 2203, 2226, 2227, 2239, 3044, 3080, 3084,
 3128, 3138, 3165, 3166, 3179, 3180, 3254,
 3265, 3288, 3292, 3319, 3323, 3405, 3484,
 3599, 3605, 3625, 3631, 3639, 3649, 3656,
 3729, 3777, 3782, 4027, 4058, 4220, 4266, 4339
 \MT@ifint 457, 2710, 3485, 3730, 4172, 4182
 \MT@ifstreq 540, 1099, 1436, 1603,
 2356, 2497, 2512, 2773, 2776, 2784, 2793,
 2999, 3229, 3573, 3693, 3715, 3717, 3895,
 3915, 3986, 4030, 4032, 4035, 4038, 4060,
 4062, 4076, 4077, 4149, 4150, 4155, 4159,
 4188, 4268, 4275, 4279, 4286, 4341, 4347, 4675
 \MT@in@clist 620,
 772, 801, 1112, 1130, 1139, 1175, 1782,
 2378, 2389, 2951, 2954, 2964, 2976, 3042, 3380
 \MT@in@rlist 648, 1158, 2487
 \MT@in@rlist@ 648
 \MT@in@rlist@@ 648
 \MT@in@tlist 636, 2438, 3053
 \MT@in@tlist@ 636
 \MT@inannotfalse 100
 \MT@inannottrue 100
 \MT@increment 733, 3852, 3859, 3866, 3872
 \MT@info 76, 91, 96, 1051, 4156, 4382, 4386
 \MT@info@missing@char 1292, 1322, 2617

- `\MT@info@nl` [76](#), [92](#), [93](#), [97](#), [812](#), [1196](#),
[1323](#), [4147](#), [4157](#), [4227](#), [4241](#), [4245](#), [4443](#),
[4447](#), [4459](#), [4466](#), [4587](#), [4624](#), [4644](#), [4649](#),
[4658](#), [4668](#), [4691](#), [4695](#), [4708](#), [4723](#), [4792](#), [4803](#)
`\MT@info@notracking` [990](#), [1100](#), [1104](#)
`\MT@info@notracking@` [1100](#), [1104](#)
`\MT@inh@do` [2535](#), [3811](#)
`\MT@inh@feat` [3767](#), [3777](#), [3780](#), [3791](#)
`\MT@inh@split` [3813](#), [3817](#)
`\MT@inlist@false` [620](#), [624](#), [637](#), [649](#), [2972](#)
`\MT@inlist@true` [620](#), [626](#), [644](#), [656](#), [661](#), [2972](#)
`\MT@is@active` [2548](#), [2723](#)
`\MT@is@char` [2555](#), [2748](#), [2757](#)
`\MT@is@charx` [2766](#), [2787](#), [2820](#)
`\MT@is@composite` [2553](#), [2807](#)
`\MT@is@feature` [771](#), [3085](#), [3293](#), [3324](#)
`\MT@is@letter` [2549](#), [2667](#), [2753](#), [2816](#)
`\MT@is@number` [2693](#), [2698](#), [2804](#)
`\MT@is@symbol` [2552](#), [2745](#)
`\MT@is@tlig` [2751](#), [2801](#)
`\MT@is@uni@comp` [2814](#), [2826](#)
`\MT@is@xchar` [2777](#), [2783](#)
`\MT@iterate` [670](#)
`\MT@kerning` [996](#), [1708](#), [4694](#)
`\MT@kerningfalse` [308](#)
`\MT@kerningtrue` [308](#)
`\MT@kn@c@name` [1716](#), [1718](#), [1761](#), [3511](#)
`\MT@kn@context` [3027](#), [3068](#)
`\MT@kn@doc@contexts` [3068](#)
`\MT@kn@factor` [311](#)
`\MT@kn@factor@` [1360](#)
`\MT@kn@inh@name` [1737](#)–[1739](#)
`\MT@kn@max` [323](#)
`\MT@kn@min` [323](#)
`\MT@kn@setname` [3302](#)
`\MT@kn@split@val` [1722](#)
`\MT@kn@unit` [317](#)
`\MT@kn@unit@` [1368](#), [1764](#)
`\MT@led@unhbox@line` [812](#)
`\MT@ledmac@setup` [806](#), [883](#)–[885](#)
`\MT@let@cn` [397](#), [400](#), [401](#), [1071](#), [1209](#),
[1507](#), [1594](#), [1640](#), [1718](#), [1933](#), [1941](#), [2355](#),
[2367](#), [2492](#), [2496](#), [2529](#), [2920](#), [3208](#), [3619](#), [3796](#)
`\MT@let@nc` .. [395](#), [1396](#), [2148](#), [2508](#), [2540](#), [2967](#),
[3937](#), [3942](#), [4036](#), [4039](#), [4282](#), [4289](#), [4706](#), [4721](#)
`\MT@let@nn`
[400](#), [1361](#), [1366](#), [1369](#), [1381](#), [1612](#), [1615](#), [1955](#)
`\MT@letterspace`
[338](#), [1811](#), [1941](#), [1943](#), [1945](#), [4756](#), [4757](#), [4759](#)
`\MT@letterspace@` [1227](#), [1297](#),
[1298](#), [1811](#), [1812](#), [1815](#), [1818](#), [1826](#), [1833](#),
[1838](#), [1846](#), [1859](#)–[1861](#), [1868](#), [1877](#), [1921](#)–
[1924](#), [2068](#), [2071](#), [2189](#), [2204](#)–[2207](#), [2240](#), [2243](#)
`\MT@letterspace@default` [338](#), [4757](#)
`\MT@listname` [2441](#),
[2489](#), [2492](#), [2499](#), [2500](#), [2502](#), [2504](#), [2525](#),
[2526](#), [2528](#), [2529](#), [2533](#), [2536](#), [3832](#), [3837](#)
`\MT@listname@count` [3602](#), [3607](#), [3609](#)
`\MT@load@inputenc` [1437](#), [1440](#), [1449](#)
`\MT@load@list` [1207](#), [1505](#), [1638](#), [1716](#), [2352](#)
`\MT@loop` [670](#), [678](#), [726](#), [3941](#)
`\MT@lower` [3158](#), [3164](#), [3564](#)
`\MT@ls@adjust` [1875](#), [2200](#)
`\MT@ls@adjust@` [2184](#), [2208](#)
`\MT@ls@adjust@empty` [2186](#), [2200](#)
`\MT@ls@adjust@relax` [2187](#), [2200](#)
`\MT@ls@basefont` [1993](#), [2002](#), [2005](#), [2006](#)
`\MT@ls@fontspec@colon` [1917](#)
`\MT@ls@fontspec@font` [1828](#), [1917](#)
`\MT@ls@outer@k` [1878](#),
[1883](#), [1891](#), [2084](#), [2098](#), [2145](#), [2163](#), [2248](#)
`\MT@ls@set@ls` [2191](#), [2194](#), [2200](#)
`\MT@ls@too@large` [2207](#), [2210](#), [4759](#)
`\MT@ls@font` [1817](#),
[1827](#), [1833](#), [1846](#), [1847](#), [1856](#), [1860](#), [1861](#),
[1866](#), [1960](#), [1971](#), [2016](#), [2018](#), [2033](#), [2034](#),
[2041](#), [2042](#), [2049](#), [2051](#), [2057](#), [2072](#), [2241](#), [2244](#)
`\MT@ltx@pickupfont` [912](#), [953](#), [958](#), [2935](#), [2943](#)
`\MT@lua` [192](#), [255](#), [353](#),
[361](#), [481](#), [518](#), [564](#), [684](#), [705](#), [2290](#), [2303](#), [2571](#)
`\MT@lua@copy@font` [1041](#)
`\MT@lua@copyfont` [1021](#), [1041](#)
`\MT@luatex@no` [253](#), [351](#)
`\MT@map@c@list@` [592](#)
`\MT@map@c@list@c` [592](#),
[1023](#), [2285](#), [2963](#), [2975](#), [2988](#), [3032](#), [3039](#),
[3081](#), [3289](#), [3320](#), [3603](#), [3778](#), [3780](#), [3790](#), [4396](#)
`\MT@map@c@list@n` [592](#), [1086](#), [2269](#), [3068](#), [3083](#), [3103](#), [3114](#),
[3152](#), [3227](#), [3275](#), [3291](#), [3322](#), [3352](#), [3427](#),
[3535](#), [3546](#), [3559](#), [3571](#), [3711](#), [3727](#), [3826](#),
[4022](#), [4025](#), [4054](#), [4056](#), [4085](#), [4167](#), [4264](#), [4337](#)
`\MT@map@t@list@` [610](#)
`\MT@map@t@list@c` ... [610](#), [639](#), [650](#), [1026](#), [1263](#),
[1581](#), [1668](#), [1739](#), [2445](#), [2994](#), [3905](#), [3954](#), [4431](#)
`\MT@map@t@list@n` [610](#), [2417](#), [2759](#), [3846](#)
`\MT@max@char` [2667](#), [2690](#)
`\MT@max@slot` [2667](#), [2716](#)
`\MT@maybe@do` .. [1081](#), [1190](#), [1497](#), [1631](#), [1709](#), [1784](#)
`\MT@maybe@gobble@with@tikz` [850](#), [1907](#), [1964](#)
`\MT@maybe@rem@from@list` [2994](#), [2998](#)
`\MT@microtype@context` [3004](#)
`\MT@minus` [1826](#), [1921](#)–[1924](#)
`\MT@MT` [2](#), [76](#), [79](#), [82](#), [84](#),
[86](#), [91](#), [92](#), [94](#), [95](#), [289](#), [783](#), [878](#), [880](#), [3641](#),
[4199](#), [4221](#), [4382](#), [4386](#), [4444](#), [4679](#), [4796](#), [4797](#)
`\MT@MT@pickupfont` [955](#), [960](#), [2935](#), [2945](#)
`\MT@next@listname` [2448](#), [2457](#)
`\MT@next@listname@` [2467](#), [2479](#), [2485](#)
`\MT@nl@ligatures` [2276](#), [3398](#)
`\MT@nl@setname` [2268](#), [3398](#)
`\MT@nofamily@false` [1187](#), [1192](#)
`\MT@nofamily@true` [1187](#), [2458](#)
`\MT@noligatures` [1000](#), [2264](#), [4768](#)
`\MT@noligatures@` [2016](#), [2018](#), [2276](#), [2279](#)
`\MT@noligatures@false` [305](#)
`\MT@noligatures@true` [305](#), [3407](#)
`\MT@nonselected@false` [1513](#), [1528](#)
`\MT@nonselected@true` [1513](#), [1516](#)
`\MT@noreset@false`
... [2676](#), [2683](#), [2692](#), [2713](#), [2773](#), [2785](#), [2793](#)
`\MT@noreset@true` [2546](#), [2676](#), [2701](#)
`\MT@old@cmd` [66](#), [71](#)–[74](#)
`\MT@opt@autofalse` [3997](#)
`\MT@opt@autotruer` [3997](#), [4087](#)
`\MT@opt@def@set` [4013](#), [4041](#), [4066](#)
`\MT@opt@DVIfalse` [3998](#)
`\MT@opt@DVIttrue` [3998](#), [4094](#), [4097](#)
`\MT@opt@expansion@false` [3996](#)
`\MT@opt@expansion@true` [3996](#)

- \MT@options 3522
 \MT@optwarn@admissible 3999, 4078, 4159
 \MT@optwarn@nan 4003, 4174, 4184
 \MT@orig@add@accent 2941
 \MT@orig@foreign@language 4809, 4811
 \MT@orig@pickupfont 2870, 2937
 \MT@orig@py@macron 950, 954, 959
 \MT@orig@select@language 4804, 4806
 \MT@outer@kern 1877, 1881, 1883, 1884, 1888, 1896,
 1987, 1988, 1990, 2145, 2146, 2163, 2221, 2260
 \MT@outer@space 1871,
 1872, 2031, 2080, 2082, 2083, 2096, 2097,
 2114, 2115, 2130, 2131, 2135, 2136, 2161
 \MT@pdf@annot 99
 \MT@pdf@or@lua 180, 188, 287, 460, 683, 4740
 \MT@pdf@tex@no 212, 350
 \MT@permute 3444, 3468, 3482, 3503, 3517, 3803, 3842
 \MT@permute@ 3842
 \MT@permute@@ 3842
 \MT@permute@@@ 3842
 \MT@permute@@@@ 3871, 3876
 \MT@permute@@@@@ 3884, 3886
 \MT@permute@define 3877, 3887–3889, 3933
 \MT@permute@reset 3846, 3939
 \MT@permutelist 3438, 3454,
 3480, 3497, 3511, 3578, 3579, 3584, 3801,
 3899, 3900, 3904, 3908, 3911, 3914–3916,
 3918–3920, 3928, 3929, 3955, 3987, 3989, 3990
 \MT@pickupfont 2935
 \MT@plain 125, 146
 \MT@pr@c@name 1207, 1209, 1465, 3438
 \MT@pr@context 1012, 3026, 3068
 \MT@pr@doc@contexts 3068
 \MT@pr@factor 311, 4183, 4460, 4461
 \MT@pr@factor@ 1360
 \MT@pr@inh@name 1261, 1262, 1264
 \MT@pr@level 311, 4458, 4459
 \MT@pr@max 323
 \MT@pr@min 323
 \MT@pr@setname 3302
 \MT@pr@split@val 1248
 \MT@pr@unit 317, 4190, 4462
 \MT@pr@unit@ 1368, 1468
 \MT@preset@aux 1470,
 1472, 1474, 1475, 1478, 1766, 1768, 1770, 1771
 \MT@preset@aux@factor 1470, 1478, 1697–1699, 1766
 \MT@preset@aux@space . 1472, 1478, 1701–1703, 1768
 \MT@preset@ex 1606, 1623
 \MT@preset@kn 1759
 \MT@preset@kn@ 1759
 \MT@preset@pr 1463
 \MT@preset@pr@ 1463
 \MT@preset@sp 1690
 \MT@preset@sp@ 1690
 \MT@ProcessOptionsWithKV 4390, 4414
 \MT@protrudechars 4315, 4319, 4458, 4646
 \MT@protrusion 991, 1189, 4465
 \MT@protrusionfalse 301
 \MT@protrusiontrue 301, 4206
 \MT@rba@expansion 764
 \MT@rba@kerning 764
 \MT@rba@protrusion 764
 \MT@rba@spacing 764
 \MT@rba@tracking 764
 \MT@register@font 1004, 2952, 3035
 \MT@register@font@cx 2987, 3035
 \MT@register@subst@font 2926, 2953, 3036
 \MT@register@subst@font@cx 2974, 3036
 \MT@rem@from@clist 631, 1036, 2957, 2982, 3000, 3527
 \MT@rem@from@list 1026, 1034
 \MT@rem@last@space 432
 \MT@repeat 670, 680, 729, 3947
 \MT@requires@latex 145, 164, 792,
 908, 1959, 2868, 4005, 4166, 4391, 4395, 4838
 \MT@requires@luatex 345, 982, 1049, 1225,
 1532, 1533, 1555, 1602, 1778, 2279, 3744,
 3750, 4090, 4320, 4334, 4476, 4553, 4561, 4640
 \MT@requires@pdftex 345,
 459, 494, 541, 807, 971, 981, 995, 999,
 1010, 1224, 1269, 1295, 1554, 1630, 1708,
 1777, 2013, 2265, 3399, 3629, 3743, 3748,
 4116, 4210, 4333, 4497, 4528, 4556, 4639, 4766
 \MT@res@a ... 554, 556, 622, 629, 632, 634, 638, 643
 \MT@res@b 555, 556, 632–634, 642, 643
 \MT@reset@context 3014, 3018, 3024, 3051
 \MT@reset@context@ 3024, 3051
 \MT@reset@ef@codes 1502, 1526, 1554, 1607
 \MT@reset@ef@codes@ 1549, 1559, 1563
 \MT@reset@kn@codes 1713, 1720, 1750
 \MT@reset@kn@codes@ 1750
 \MT@reset@pr@codes 1204, 1211, 1220
 \MT@reset@pr@codes@ 1220
 \MT@reset@sp@codes 1635, 1642, 1680
 \MT@reset@sp@codes@ 1680
 \MT@restore@catcodes 5, 7, 8, 159, 165, 4844
 \MT@restore@p@h 843, 870, 925
 \MT@saved@setupfont 4379, 4387
 \MT@scale 740, 1272, 1334, 1348, 1351, 1568
 \MT@scale@factor 1274, 1305, 1332, 1480, 1625
 \MT@scale@to@em .. 1251, 1257, 1269, 1302, 1486,
 1648, 1655, 1662, 1726, 1733, 1944, 2064, 2235
 \MT@scrubfeature 1062, 1074, 3550
 \MT@scrubfeatures 1074, 3145, 3576
 \MT@selectedfalse 304
 \MT@selectedtrue 304
 \MT@series 1058, 1174, 2461, 2473
 \MT@set@all@ex 1549, 1626
 \MT@set@all@kn 1750, 1772
 \MT@set@all@pr 1213, 1220, 1476
 \MT@set@all@sp 1680, 1705
 \MT@set@babel@context 4781, 4807, 4812, 4826
 \MT@set@codes . 1210, 1231, 1508, 1641, 1719, 2368
 \MT@set@curr@ls 1868
 \MT@set@curr@ok 1888, 1890, 1896, 1899, 2119, 2123
 \MT@set@curr@os 1870, 1898, 2119, 2123
 \MT@set@default@set 3320, 3325, 3333
 \MT@set@ex@codes 1530, 4571, 4573
 \MT@set@ex@codes@n 1513, 1530, 4573
 \MT@set@ex@codes@es 1498, 4571
 \MT@set@ex@heirs 1581, 1618
 \MT@set@inh@list 3774, 3776
 \MT@set@inputenc 1206, 1427, 1504, 1637, 1715, 2534
 \MT@set@inputenc@ 1430, 1432
 \MT@set@kn@codes 1710
 \MT@set@kn@heirs 1739, 1743
 \MT@set@listname 1208,
 1359, 1506, 1592, 1639, 1717, 1930, 2829
 \MT@set@ls@basefont 2002
 \MT@set@ls@font 1865, 1958, 1971

- \MT@set@named@keys 3440, 3456, 3478, 3499, 3513, 3522
 \MT@set@pr@codes 1191
 \MT@set@pr@heirs 1265, 1455
 \MT@set@sp@codes 1632
 \MT@set@sp@heirs 1668, 1672
 \MT@set@tr@codes 1794, 1966, 1976
 \MT@set@tr@zero 1813, 2002, 4361
 \MT@SetExpansion 3452, 3454
 \MT@SetExtraKerning 3509, 3511
 \MT@SetExtraSpacing 3495, 3497
 \MT@SetProtrusion 3436, 3438
 \MT@SetTracking 3474, 3476
 \MT@setup@ 790, 794, 4108, 4109, 4839
 \MT@setup@contexts 3013, 3031
 \MT@setup@copies 4430, 4446
 \MT@setup@expansion 4433, 4470
 \MT@setup@kerning 4437, 4686, 4729
 \MT@setup@noligatures 4438, 4764
 \MT@setup@PDF 4429, 4441, 4451
 \MT@setup@protrusion 4432, 4454
 \MT@setup@spacing 4436, 4653, 4730
 \MT@setup@spacing@check 4671, 4842, 4843
 \MT@setup@tracking 4434, 4638, 4701, 4728
 \MT@setup@wartracking 4435, 4733
 \MT@setupfont .. 967, 2929, 4380, 4383, 4387, 4423
 \MT@setupfont@hook 858, 888, 894, 899, 925, 932, 970
 \MT@shape 1058, 1174, 2462, 2474
 \MT@shorthandoff 4790, 4821, 4824
 \MT@show@pdfannot 106
 \MT@shrink 320, 1521,
 4493, 4494, 4501, 4502, 4505, 4509, 4576, 4589
 \MT@shrink@ 1521, 1536, 1540, 1546, 1591
 \MT@shrink@default 336
 \MT@size 655, 659, 660, 1050, 1058, 1174
 \MT@size@name 648, 2489
 \MT@S@L@doword 943, 944
 \MT@sp@cc@name 1638, 1640, 1692, 3497
 \MT@sp@context 3027, 3068, 4675
 \MT@sp@doc@contexts 3068
 \MT@sp@factor 311
 \MT@sp@factor@ 1360
 \MT@sp@inh@name 1666–1668
 \MT@sp@max 323
 \MT@sp@min 323
 \MT@sp@setname 3302
 \MT@sp@split@val 1644
 \MT@sp@unit 317
 \MT@sp@unit@ 1368, 1695
 \MT@spacing 996, 1629, 4667
 \MT@spacingfalse 307
 \MT@spacingtrue 307
 \MT@split@codes 1233, 1237
 \MT@split@name 975, 1058
 \MT@step 320, 1522, 4496,
 4498, 4516, 4520, 4523, 4590, 4593, 4594, 4598
 \MT@step@ 1522, 1536, 1540, 1546, 1591
 \MT@stretch 320, 1520, 4490,
 4491, 4494, 4501, 4503, 4508, 4511, 4575, 4589
 \MT@stretch@ 1520, 1536, 1540, 1546, 1591
 \MT@stretch@default 336, 4491
 \MT@strip@prefix 2766, 2787, 2819
 \MT@temp ... 1215–1218, 1682–1686, 1752–1755,
 2036, 2040, 2043, 2051, 2054, 2111, 2114,
 2120, 2124, 2127, 2130, 2135, 2143, 2145,
 2151, 2225, 2228, 2245, 3248, 3273, 4394,
 4399, 4404, 4405, 4407, 4410, 4605, 4609, 4613
 \MT@tempencoding 3879, 3880, 3890, 3896
 \MT@tempfamily 3891
 \MT@tempseries 3892
 \MT@tempshape 3893
 \MT@tempsize 3847, 3894, 3903, 3905, 3909
 \MT@test@east 2062, 2233, 3127, 3136
 \MT@text@microtypecontext 3021
 \MT@textls 49, 2186, 2187, 2189
 \MT@textmicrotypecontext 3021
 \MT@the@pr@code 1222, 1252, 1258, 1862
 \MT@the@pr@code@tr 1222, 1862
 \MT@tikz@setup 850, 871, 948
 \MT@tlist@break 610, 645, 667, 2450, 3993
 \MT@toks 299,
 1323, 1354, 1587, 2547, 2582, 2609, 2681,
 2688, 2738, 2779, 2794, 2803, 2834, 2846, 2854
 \MT@tr@cc@name 1932,
 1933, 1940, 1941, 1954, 1955, 3479, 3486
 \MT@tr@context 3027, 3068
 \MT@tr@doc@contexts 3068
 \MT@tr@factor@ 1931
 \MT@tr@font@list 1780
 \MT@tr@ispace 1836, 1837, 1948
 \MT@tr@ligatures 1852, 1951, 2015, 2018
 \MT@tr@max 323, 2211–2213
 \MT@tr@min 323, 2215–2217
 \MT@tr@noligatures 1852, 2012
 \MT@tr@okern 1850, 1950
 \MT@tr@ospace 1839, 1840, 1948
 \MT@tr@outer@icr 2124, 2153
 \MT@tr@outer@icr@ 2153
 \MT@tr@outer@l 1873, 2076
 \MT@tr@outer@next 2104, 2108, 2155
 \MT@tr@outer@r 1900, 2104, 2120, 2127, 2153
 \MT@tr@outer@r@ 2105, 2110
 \MT@tr@set@okern 1851, 2221
 \MT@tr@set@okern@ 2226, 2227, 2232
 \MT@tr@set@space 1842, 2032
 \MT@tr@set@space@ 2037–2039, 2045
 \MT@tr@set@space@@ 2048, 2053, 2056, 2061
 \MT@tr@setname 3302
 \MT@tr@unit@ 1932, 2063, 2234
 \MT@tracking 983, 1780, 1966, 2931, 4361, 4362, 4648
 \MT@tracking@ 1780, 4362
 \MT@trackingfalse 309, 4705
 \MT@trackingtrue 309
 \MT@try@order 2445, 2453
 \MT@undefined@char 2728, 2733, 2744
 \MT@upper 3158, 3164, 3564
 \MT@use@set 3289, 3294, 3302
 \MT@UseMicrotypeSet 3285, 3287
 \MT@val 3044–3046, 3048, 3054, 3056,
 3059, 3115, 3119, 3127, 3130, 3133, 3153–
 3155, 3167, 3170, 3172, 3175, 3177, 3178,
 3184, 3186, 3187, 3189, 3192, 3194, 3205,
 3208, 3212, 3215–3217, 3219, 3221, 3228–
 3230, 3233, 3242–3244, 3256, 3258, 3263,
 3265, 3267, 3269, 3536, 3538, 3547, 3550,
 3552, 3560–3562, 3572–3574, 3580, 3585,
 3825, 3832, 3836, 3837, 4014, 4015, 4018,
 4026, 4027, 4030, 4032, 4035, 4038, 4057,
 4058, 4060, 4062, 4265, 4266, 4268, 4275,
 4279, 4286, 4292, 4338, 4339, 4341, 4347, 4350

- `\MT@variants` 2438, [3344](#), [3355](#)
`\MT@vinfo` [76](#), [93](#), [989](#), [1101](#), [1105](#), [1363](#),
[1372](#), [1376](#), [1411](#), [1595](#), [1613](#), [1797](#), [1815](#),
[2293](#), [2304](#), [2361](#), [2385](#), [2395](#), [2398](#), [2502](#),
[2504](#), [2510](#), [3025](#), [3048](#), [3382](#), [3388](#), [4147](#),
[4154](#), [4271](#), [4277](#), [4283](#), [4290](#), [4344](#), [4349](#), [4783](#)
`\MT@warn@ascii` [2690](#), [2833](#)
`\MT@warn@axis@empty` [3255](#), [3266](#), [3271](#)
`\MT@warn@code@tootlarge` [1338](#), [1341](#), [1345](#)
`\MT@warn@err` [76](#), [4151](#), [4152](#)
`\MT@warn@ex@tootlarge` [1571](#), [1574](#), [1585](#)
`\MT@warn@maybe@inputenc` ... [2283](#), [2847](#), [2855](#), [2858](#)
`\MT@warn@nodim` [3683](#), [3696](#), [3706](#), [3720](#)
`\MT@warn@number@tootlarge` [2717](#), [2839](#)
`\MT@warn@preset@towidth` ... [1469](#), [1489](#), [1696](#), [1765](#)
`\MT@warn@rest` [2588](#), [2625](#), [2844](#)
`\MT@warn@tracking@DVI` [1816](#), [4735](#)
`\MT@warn@unknown` ... [2578](#), [2594](#), [2604](#), [2629](#), [2852](#)
`\MT@warn@unknown@once` [2870](#), [2871](#)
`\MT@warning` . [67](#), [76](#), [94](#), [2212](#), [2216](#), [2296](#), [2513](#),
[3102](#), [3131](#), [3219](#), [3272](#), [3367](#), [3390](#), [3487](#),
[3616](#), [3685](#), [3733](#), [3746](#), [3751](#), [3897](#), [3918](#),
[3989](#), [4151](#), [4156](#), [4253](#), [4330](#), [4369](#), [4371](#), [4372](#)
N `newpx` (package) [21](#), [59](#), [144](#), [236](#)
`newtx` (package) [21](#), [59](#), [144](#), [236](#)
`newunicodechar` (package) [95](#), [236](#)
O `\outputmode` [194](#), [4090](#)
P `\pdfadjustinterwordglue` [4363](#), [4657](#)
`\pdfadjustspacing` [196](#), [944](#), [4322](#), [4326](#), [4618](#)
`\pdfappendkern` [4365](#), [4690](#)
`\pdfcopyfont` [1020](#)
`\pdfcprot` (package) [26](#), [30](#), [54](#), [64](#), [99](#), [228](#)
`\pdffeedback` [1049](#)
`\pdffontexpand` [1533](#), [1536](#), [1540](#), [1546](#)
`\pdffontsize` [1048](#), [1049](#)
`\pdfnoligatures` [2279](#), [2295](#), [2301](#)
`\pdfprependkern` [4364](#), [4689](#)
`\pdfprotrudechars` [195](#), [4321](#), [4325](#), [4752](#)
`pdfTeX` (engine) . [1](#), [4](#), [6–9](#), [13–19](#), [23–28](#), [30–35](#),
[37–39](#), [41–43](#), [45](#), [46](#), [52](#), [53](#), [57](#), [58](#), [64](#), [70](#),
[72–75](#), [79](#), [80](#), [83](#), [85](#), [112](#), [114](#), [116](#), [118](#),
[128](#), [132](#), [133](#), [135](#), [136](#), [139](#), [195](#), [198](#), [228–234](#)
`pdftexcmds` (package) [37](#), [234](#)
`\pdftracingfonts` [355](#)
`\pickup@font` [2901](#), [2903](#), [2912](#), [2935–2937](#)
`pifont` (package) [99](#)
`pinyin` (package) [56](#), [233](#), [234](#)
`\protrudechars` [195](#), [4321](#)
`protrusion` (option) [6](#), [11](#), [124](#), [125](#)
`psnfss` (package) [79](#), [197](#), [236](#)
`psstricks` (package) [133](#)
`pxfonts` (package) [21](#), [144](#)
Q `qfonts` (package) [21](#), [144](#), [230](#)
R `ragged2e` (package) [26](#), [81](#), [138](#), [234](#), [235](#)
`reledmac` (package) [53](#), [236](#)
`reysize` (package) [107](#)
S `selected` (option) [8](#), [14](#), [33](#), [69](#), [90](#), [125](#), [229](#)
`\selectfont` [1967](#), [3017](#), [3029](#), [4258](#), [4834](#)
`\set@fontsize` [3212](#)
`\SetExpansion` [14](#), [36](#), [3449](#)
`\SetExtraKerning` [18](#), [38](#), [3507](#)
`\SetExtraSpacing` [19](#), [39](#), [3492](#)
`\SetProtrusion` [13](#), [35](#), [3433](#)
`\SetTracking` [15](#), [37](#), [3472](#)
`\sfcode` [4674](#)
`shapepar` (package) [29](#), [53](#)
`\shbscode` [1663](#), [1664](#), [1675](#), [1678](#), [1685](#)
`\showhyphens` [4604](#)
`shrink` (option) .. [8](#), [15](#), [33](#), [127](#), [135](#), [137](#), [231](#), [233](#)
T `\tagcode` [1856](#), [2282](#), [2288](#)
`tempora` (package) [21](#), [144](#), [236](#)
`tex4ht` (package) [54](#), [232](#)
`TeX Live` (distribution) [28](#), [39](#), [230](#)
`\textls` [23](#), [52](#), [53](#), [917](#), [941](#), [1876](#), [1962](#), [1979](#), [2184](#)
`\MT@warning@nl` [76](#), [88](#), [95](#), [158](#), [276](#), [782](#), [786](#), [822](#),
[834](#), [901](#), [1315](#), [1352](#), [1404](#), [1444](#), [1490](#),
[1586](#), [1800](#), [2024](#), [2834](#), [2840](#), [2845](#), [2853](#),
[2902](#), [3459](#), [3658](#), [4000](#), [4007](#), [4018](#), [4100](#),
[4152](#), [4157](#), [4191](#), [4198](#), [4234](#), [4248](#), [4420](#),
[4477](#), [4521](#), [4577](#), [4596](#), [4660](#), [4676](#), [4741](#), [4828](#)
`\MT@while@num` [676](#), [700](#), [706](#), [928–930](#)
`\MT@with@babel@and@T`
... [799](#), [860–862](#), [890–892](#), [4816–4820](#), [4823](#)
`\MT@with@package@T` [796](#), [847](#), [849](#),
[864](#), [867–869](#), [871](#), [876](#), [883–887](#), [897](#), [909](#),
[923](#), [924](#), [926](#), [948](#), [949](#), [2870](#), [2871](#), [2913](#), [4659](#)
`\MT@xadd` [104](#), [576](#), [2386](#), [2391](#), [2396](#),
[2399](#), [2979](#), [2990](#), [3056](#), [3156](#), [3384](#), [3563](#), [3832](#)
`\MT@xaddb` [584](#), [3583](#), [3907](#)
`\MT@xdef@n` .. [391](#), [3304](#), [3307](#), [3336](#), [3341](#), [3370](#),
[3486](#), [3578](#), [3600](#), [3762](#), [3798](#), [3928](#), [4015](#), [4017](#)
`\MT@xetex@no` [244](#)
`\MT@xspace` [2143](#), [2158](#)
`\MT@xspace@` [2158](#)
`\MT@xunicode@false` [846](#)
`\MT@xunicode@true` [846](#), [847](#), [886](#)
`\normalfont` [966](#), [4616](#), [4620](#)
`\nullfont` [966](#)

trace (package)	101, 232	\tracingmicrotypeinpdfalltrue	105
\tracingmicrotype	87	tracking (option)	6, 11, 23, 31, 125
\tracingmicrotypeinpdf	98	txfonts (package)	21, 144
\tracingmicrotypeinpdfallfalse	105		
U ucs (package)	95, 233	\UseMicrotypeSet	12, 33, 60, 73, 3096, 3283
ulgothic (package)	21	\UTF@four@octets@noexpand	2731
\UnicodeEncodingName	2812	\UTF@three@octets@noexpand	2730
unit (option)	7, 32, 230, 232	\UTF@two@octets@noexpand	2729
V verbose (option)	9, 31, 32, 127, 229, 230, 233		
W wordcount (package)	52, 233		
X XCharter (package)	21, 144, 236	\XeTeXglyphname	2621
xeCJK (package)	27, 99, 100, 235, 236	\XeTeXlastfontchar	729
X _Y TeX (engine)	1, 4, 6, 7, 9, 12, 13, 20, 28–30, 34, 38, 40, 50, 53, 56, 59, 83, 92, 132, 137, 139, 143, 146, 149, 157, 235–237	\XeTeXprotrudechars	4329
\XeTeXcharglyph	2615	\xspace	2142, 2143
\XeTeXfonttype	2566	xspace (package)	82, 235
\XeTeXglyph	1286	\xspaceskip	2090, 2093
\XeTeXglyphindex	2602	xunicode (package)	96, 97, 235
Y yfonts (package)	24	xunicode-addon (package)	99
Z zefonts (package)	21, 143		

E The L^AT_EX Project Public License

LPPL Version 1.3c 2008-05-04

Copyright 1999, 2002–2008 L^AT_EX3 Project

Preamble

The L^AT_EX Project Public License (LPPL) is the primary license under which the L^AT_EX kernel and the base L^AT_EX packages are distributed.

You may use this license for any work of which you hold the copyright and which you wish to distribute. This license may be particularly suitable if your work is T_EX-related (such as a L^AT_EX package), but it is written in such a way that you can use it even if your work is unrelated to T_EX.

The section ‘WHETHER AND HOW TO DISTRIBUTE WORKS UNDER THIS LICENSE’, below, gives instructions, examples, and recommendations for authors who are considering distributing their works under this license.

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.

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the L^AT_EX3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document ‘c_fgguide.tex’ and ‘modguide.tex’ in the base L^AT_EX distribution for suggestions.

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 us-

ing installation facilities provided by the Work, transformations of the Work, copying of components of the Work, or other activities. Note that modification of any installation facilities provided by the Work constitutes modification of the Work.

Current Maintainer: A person or persons nominated as such within the Work. If there is no such explicit nomination then it is the ‘Copyright Holder’ under any applicable law.

Base Interpreter: A program or process that is normally

needed for running or interpreting a part or the whole of the Work.

A Base Interpreter may depend on external components but these are not considered part of the Base Interpreter provided that each external component clearly identifies itself whenever it is used interactively. Unless explicitly specified when applying the license to the Work, the only applicable Base Interpreter is a ‘L^AT_EX-Format’ or in the case of files belonging to the ‘L^AT_EX-format’ a program implementing the ‘T_EX language’.

Conditions on Distribution and Modification

1. Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
2. You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.
3. You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
4. If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Maintainer are considered to be updated versions of the Work.
5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
 - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
 - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
 - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
 - (d) You distribute at least one of the following with the Derived Work:
 - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
 - ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.
10. (a) A Derived Work may be distributed under a different license provided that license itself honors

the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.

- (b) If a Derived Work is distributed under a different license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the

Work.

11. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.
12. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless required by applicable law or agreed to in writing will The Copyright Holder, or any au-

thor named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

Maintenance of The Work

The Work has the status ‘author-maintained’ if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that it is ‘author-maintained’.

The Work has the status ‘maintained’ if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status ‘maintained’ to ‘unmaintained’ if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.
2. If this search is successful, then enquire whether the Work is still maintained.
 - (a) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.
 - (b) If the search is unsuccessful or no action to resume active maintenance is taken by the Current

Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a L^AT_EX work, this could be done, for example, by posting to `comp.text.tex`.)

3. (a) If the Current Maintainer is reachable and agrees to pass maintenance of the Work to you, then this takes effect immediately upon announcement.
 - (b) If the Current Maintainer is not reachable and the Copyright Holder agrees that maintenance of the Work be passed to you, then this takes effect immediately upon announcement.
4. If you make an ‘intention announcement’ as described in 2b above and after three months your intention is challenged neither by the Current Maintainer nor by the Copyright Holder nor by other people, then you may arrange for the Work to be changed so as to name you as the (new) Current Maintainer.
5. If the previously unreachable Current Maintainer becomes reachable once more within three months of a change completed under the terms of 3b or 4, then that Current Maintainer must become or remain the Current Maintainer upon request provided they then update their communication data within one month.

A change in the Current Maintainer does not, of itself, alter the fact that the Work is distributed under the LPPL license.

If you become the Current Maintainer of the Work, you should immediately provide, within the Work, a prominent and unambiguous statement of your status as Current Maintainer. You should also announce your new status to the same pertinent community as in 2b above.

Whether and How to Distribute Works under This License

This section contains important instructions, examples, and recommendations for authors who are considering distributing their works under this license. These authors are addressed as ‘you’ in this section.

Choosing This License or Another License

If for any part of your work you want or need to use *distribution* conditions that differ significantly from those in this license, then do not refer to this license anywhere in your work but, instead, distribute your work under a different license. You may use the text of this license as a model for your own license, but your license should not refer to the LPPL or otherwise give the impression that your work is distributed under the LPPL.

The document ‘modguide.tex’ in the base L^AT_EX distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing L^AT_EX under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated to L^AT_EX, the discussion in ‘modguide.tex’ may still be relevant, and authors intending to distribute their works under any license are encouraged to read it.

A Recommendation on Modification Without Distribution

It is wise never to modify a component of the Work, even for your own personal use, without also meeting the above conditions for distributing the modified component. While you might intend that such modifications will never be distributed, often this will happen by accident – you may forget that you have modified that component; or it may not occur to you when allowing others to access the modified version that you are thus distributing it and violating the conditions of this license in ways that could have legal implications and, worse, cause problems for the community. It is therefore usually in your best interest to keep your copy of the Work identical with the public one. Many works provide ways to control the behavior of that work without altering any of its licensed components.

How to Use This License

To use this license, place in each of the components of your work both an explicit copyright notice including your name and the year the work was authored and/or last substantially modified. Include also a statement that the distribution and/or modification of that component is constrained by the conditions in this license.

Here is an example of such a notice and statement:

```
%% pig.dtx
%% Copyright 2005 M. Y. Name
%
% This work may be distributed and/or modified under the
% conditions of the LaTeX Project Public License, either version 1.3
% of this license or (at your option) any later version.
% The latest version of this license is in
% https://www.latex-project.org/lppl.txt
% and version 1.3 or later is part of all distributions of LaTeX
% version 2005/12/01 or later.
%
% This work has the LPPL maintenance status ‘maintained’.
%
% The Current Maintainer of this work is M. Y. Name.
%
% This work consists of the files pig.dtx and pig.ins
% and the derived file pig.sty.
```

Given such a notice and statement in a file, the conditions given in this license document would apply, with the ‘Work’ referring to the three files ‘pig.dtx’, ‘pig.ins’, and ‘pig.sty’ (the last being generated from ‘pig.dtx’ using ‘pig.ins’), the ‘Base Interpreter’ referring to any ‘L^AT_EX-Format’, and both ‘Copyright Holder’ and ‘Current Maintainer’ referring to the person ‘M. Y. Name’.

If you do not want the Maintenance section of LPPL to apply to your Work, change ‘maintained’ above into ‘author-maintained’. However, we recommend that you use ‘maintained’ as the Maintenance section was added in order to ensure that your Work remains useful to the community even when you can no longer maintain and support it yourself.

Derived Works That Are Not Replacements

Several clauses of the LPPL specify means to provide reliability and stability for the user community. They therefore concern themselves with the case that a Derived Work is intended to be used as a (compatible or incompatible) replacement of the original Work. If this is not the case (e.g., if a few lines of code are reused for a completely different task), then clauses 6b and 6d shall not apply.

Important Recommendations

Defining What Constitutes the Work

The LPPL requires that distributions of the Work contain all the files of the Work. It is therefore important that you provide a way for the licensee to determine which files constitute the Work. This could, for example, be achieved by explicitly listing all the files of the Work near the copyright notice of each file or by using a line such as:

```
% This work consists of all files listed in manifest.txt.
```

in that place. In the absence of an unequivocal list it might be impossible for the licensee to determine what is considered by you to comprise the Work and, in such a case, the licensee would be entitled to make reasonable conjectures as to which files comprise the Work.