MinionPro Support for \LaTeX

Achim Blumensath Andreas Bühmann Michael Zedler

v2.1 – 2007/03/15

Contents

1 Overview .................................................. 2
2 Interference with other packages .............................................. 2
3 Options .......................................................... 2
4 Figure selection .................................................. 4
5 Additional font shapes and symbols .......................................... 5
6 Language support .................................................................. 6
7 Searching for figures or for words containing ligatures in PDF documents ........................................... 6
8 NFSS classification ................................................................ 7
9 Version history ..................................................................... 7
10 The main style file .......................................................... 8
  10.1 Options .................................................................. 8
  10.2 Font declarations ...................................................... 11
  10.3 Font selection .......................................................... 13
  10.4 Greek letters ........................................................... 13
  10.5 pdf\LaTeX\ to-unicode support .................................... 15
  10.6 Superior and inferior figures ........................................ 17
  10.7 Additional symbols ..................................................... 20
  10.8 Integral symbols ....................................................... 21
  10.9 Open G support ........................................................ 22
  10.10 Logos .................................................................. 22
  10.11 AMS .................................................................. 23
11 Support for character protrusion ............................................. 23
12 Font definition files .......................................................... 28
1 Overview

The MinionPro package provides support for the MinionPro font family from Adobe. You can use these fonts in a \LaTeX document by adding the command

\usepackage{MinionPro}

to the preamble. This will change both the text font and the math font to MinionPro. If you prefer another math font (such as eulerm) use the option onlytext as explained in Section 3.

2 Interference with other packages

The MinionPro package automatically loads the following packages: textcomp, amsmath, and MnSymbol (version 1.4). If you want to pass options to these packages you can either put the corresponding \usepackage command before the \usepackage{MinionPro} or you can include the options in the \documentclass command. The MinionPro package is not compatible with amssymb and amsfonts. Please see also the corresponding section in the MnSymbol documentation.

The MinionPro package includes support files for the microtype package (version 1.8 or higher), consult the package's documentation for further details.

There is also a slight incompatibility with the dcolumn package which expects all figures to have the same width. If you want to use this package you either have to specify the mathtabular option (this is the brute force solution, not recommended), or you can use the figureversion{tabular} command to switch to tabular figures in front of every table (much better, but also more work). In addition, dcolumn sets figures in math mode, hence the choice of math figures (see Section 3) determines if text or lining figures are used.

3 Options

Font selection

The following options specify which version of the fonts you want to use. The default settings are marked with an asterisk*.

- smallfamily* use only regular and bold face
- medfamily use semibold face in addition to smallfamily
- fullfamily use medium face in addition to medfamily
- noopticals* use only the optical size Text
- opticals use the optical sizes Caption, Text, Subhead, and Display
- slides use only the optical size Caption (useful for slides)
- normalsize* adapt optical sizes to the normal font size (10 pt, 11 pt, 12 pt)
- nonormalsize use static settings for the optical sizes

Since MinionPro comes in only four different optical sizes we use a variable mapping from font size to the optical size. This means that, both for 10 pt and 11 pt documents,
text set in `\small` size will use the Caption size. Sometimes it might be desirable to turn off this automatism – for instance, if you want to load the \text{MinionPro} package before the `\documentclass` command. In these cases you can use the `nonormalsize` option to do so.

The package also provides a way to only change the text fonts or only the math fonts.

onlytext  only change the text fonts
onlymath  only change the math fonts

Figure selection

MinionPro offers four different figure versions. A detailed description is given in Section 4.

The default version can be selected by the following options:

textosf  use text figures in text mode
mathosf  use text figures in math mode
osf*    use text figures in text and math mode
textlf  use lining figures in text mode
mathlf  use lining figures in math mode
lf  use lining figures in text and math mode
mathtabular  use tabular figures in math mode

Calligraphic fonts

These options specify which font is used by the `\mathcal` command.

mnsy*  use the calligraphic font from MnSymbol: $ABC$
cmsy  take the calligraphic symbols from Computer Modern: $ABC$
swash  use the swash capitals from MinionPro: $ABC$
abx  use the calligraphic symbols provided by mathabx: $ABCabc$

(This font contains also lowercase letters, but it is not quite finished.)

Blackboard bold letters

You can also select different fonts for the `\mathbb` command.

amsbb*  use the AMS blackboard font: $\mathbb{NZQRC}$
fourierbb  use the Fourier blackboard font: $\mathbb{NZQRC}$
lucidabb  use the (commercial) Lucida Math blackboard font: $\mathbb{NZQRC}$

Greek letters

The following options specify whether you want to use upright or italic Greek letters in math mode.

mixedgreek*  uppercase Greek is upright, lowercase Greek is italic
italicgreek  all Greek letters are italic
frenchmath  all Greek letters and the uppercase Roman letters are upright
Upright and italic Greek letters are also directly accessible via the commands \upgamma, \itgamma, \upGamma, \itGamma, etc.

**Miscellaneous options**

- **minionint**
  - take the integral symbols from MinionPro, not from MnSymbol:
    \[ \int \] instead of \[ \int \]

- **openg**
  - use \( \partial \) instead of \( g \) in math mode.

- **loosequotes**
  - The quote signs of MinionPro are set rather tight. This can lead to undesirable spacing for apostrophes. The loosequotes option slightly increases the side bearings of quotes.
  - This option requires pdf\TeX\ 1.40 and microtype 2.0. Beware that this option prevents hyphenation of words containing apostrophes. Such words will require explicit hyphenation commands \-.

- **footnotefigures**
  - use special figures for footnote marks, i.e., example\textsuperscript{6,9} instead of example\textsuperscript{6,9}. This option can only be used if the footnote marks consist solely of figures.

4 Figure selection

MinionPro offers four different figure versions. One can choose between text figures (lowercase figures) and lining figures (uppercase figures) and one can choose between proportional figures (figures with different widths) and tabular figures (all figures have the same width, useful mainly for tables).

<table>
<thead>
<tr>
<th></th>
<th>text figures</th>
<th>lining figures</th>
</tr>
</thead>
<tbody>
<tr>
<td>proportional</td>
<td>0123456789</td>
<td>0123456789</td>
</tr>
<tr>
<td>tabular</td>
<td>0123456789</td>
<td>0123456789</td>
</tr>
</tbody>
</table>

The \texttt{\figureversion} command can be used to switch between different figure versions. Possible parameters are:

- \texttt{text, osf} text figures
- \texttt{lining, lf} lining figures
- \texttt{tabular, tab} tabular figures
- \texttt{proportional, prop} proportional figures

Usually it is desirable to set most text with proportional figures and to use tabular figures only in tables and lists. Unfortunately most \LaTeX\ document classes do not support fonts with several figure versions. Therefore we provide a package \texttt{tab\figures} that patches some common document classes and packages (the standard \LaTeX\ classes, KOMA-Script, memoir, and amsmath) to use tabular figures at some places. The \texttt{tab\figures} package supports the following options:
5 Additional font shapes and symbols

In addition to the normal small caps shape \textsc{} there is a letterspaced version called \textssc{}. It is accessible via the commands \textsscshape{} and \textscshape{}. In order to use the \textssc{} shape throughout your document specify \renewcommand{\textscshape}{\textsscshape} in the preamble of your document.

Swash capitals like 'Canadian Mountain Holidays' are accessed via the \textsw{} fontshape{} and the commands \textswshape{} and \textswshape{}.

\textsc{} \begin{quote} This is a Sample Text \end{quote} \textssc{} \begin{quote} This is a Sample Text \end{quote} \textsw{} \begin{quote} This is a Sample Text \end{quote}

The \texttt{MinionPro} package provides all symbols from the \texttt{MnSymbol} package. Additionally, the following math symbols are available:

\begin{verbatim}
\digamma  \varkappa    \varbeta
\backepsilon \varbackepsilon  \hbar
\hslash  \lambda \varlambda \varbackslash
\jmath  \eth \kappa \Bbbk
\slashedzero \greek{l} \openg
\end{verbatim}

Small and slanted fractions are fractions with a height matching the font’s body size. These are useful for typesetting, e.g., $\cos(\frac{1}{2}x + \frac{3}{2}y)$ or "$\frac{1}{2}$ litres of red wine" and can be accessed via

\begin{verbatim}
\smallfrac{\langle \texttt{numerator} \rangle}{\langle \texttt{denominator} \rangle} \frac{1}{2}
\slantfrac{\langle \texttt{numerator} \rangle}{\langle \texttt{denominator} \rangle} \frac{3}{7}
\end{verbatim}

Note that only figures can be used for \langle \texttt{numerator} \rangle and \langle \texttt{denominator} \rangle.

Ornaments can be accessed via the \texttt{pifont} package with the command

\begin{verbatim}
\Pisymbol{MinionPro-Extra}{\langle \texttt{number} \rangle}
\end{verbatim}

The available glyphs are listed in the table below. Version 1.000 of the MinionPro font provides only ornaments 100–122.
The following encodings are supported:

<table>
<thead>
<tr>
<th>Latin</th>
<th>Cyrillic</th>
<th>Greek</th>
</tr>
</thead>
<tbody>
<tr>
<td>OT1, T1, TS1, LY1, T5</td>
<td>T2A, T2B, T2C, X2, OT2</td>
<td>LGR (to be used with babel, including polytonikogreek), LGI (Ibycus transliteration scheme)</td>
</tr>
</tbody>
</table>

In order to typeset Greek text with the Ibycus transliteration scheme, specify

\usepackage[ibycus,⟨otherlanguages⟩]{babel}

in the preamble and consult the documentation given in ibycus-babel.pdf on ctan. \setgreekfontsize is not supported.

7 Searching for figures or for words containing ligatures in PDF documents

Searching for figures or for words containing ligatures in PDF documents may not be possible depending on the way the PDF file was created. The following table gives an overview of which glyphs may cause problems.
To make figures and ligatures searchable when using \pdfTeX 1.40, you need to enable glyph-to-unicode translation and load the default mapping table:

\input glyphtounicode
\pdfgentounicode=1

See the \pdfTeX manual for details.

### 8 NFSS classification

Parenthesised combinations are provided via substitutions.

<table>
<thead>
<tr>
<th>encoding</th>
<th>family</th>
<th>series</th>
<th>shape</th>
</tr>
</thead>
<tbody>
<tr>
<td>OT1, T1, TS1, LY1, T5</td>
<td>MinionPro-OsF, MinionPro-LF, MinionPro-TOsF, MinionPro-TLF</td>
<td>m, b (sb, bx), eb</td>
<td>n, it (sl), sw(^1), sc, scit (scssl, scsw), ssc, scit (sscs, sscss)</td>
</tr>
<tr>
<td>LGR, LGI, T2A, T2B, T2C, X2, OT2</td>
<td>MinionPro-OsF, MinionPro-LF, MinionPro-TOsF, MinionPro-TLF</td>
<td>m, b (sb, bx), eb</td>
<td>n, it (sl)</td>
</tr>
<tr>
<td>OML</td>
<td>MinionPro-TOsF</td>
<td>m, b (sb, bx), eb</td>
<td>n, it</td>
</tr>
<tr>
<td>u</td>
<td>MinionPro-Extra</td>
<td>m, b (sb, bx), eb</td>
<td>n, it (sl)</td>
</tr>
</tbody>
</table>

\(^1\)via substitution in ts1 encoding

### 9 Version history

Version 2.0: Initial Release on CTAN
Version 2.1:
• added package options onlytext and onlymath
• added package option loosequotes
• added package option openg
• added package options normalsize and nonormalsize
• fixed package option frenchmath
• fixed package option abx
• added support for pdfTEX 1.4 CMAP inclusion
• update to microtype version 1.8
• added \tabfigures to automatically handle tabular figures in toc, equation labels, bibliographies, enumerations
• fixed \text accent
• fixed \text accent in OT1 encoding
• fixed slashed zero in font version 2.000
• fixed arrows in TSI and U encodings
• fixed LGR and LG1 encodings to use \phi instead of \Phi
• fixed \text p in LG1 encoding
• added punctuation support in LG1 encoding (thanks to Jens Boerstinghaus)
• added symbols \hslash, \lambdabar, \lambdaslash
• fixed side bearings of \sigma in math mode
• added CODINGSCHEME statements to encoding files
• fixed usage of MnSymbols "$" in doc.sty's module prefix
• reduce number of raw encodings to five per font

10 The main style file

10.1 Options

\newcommand\Mn@minionint@opticals{-NoOpticals}
\newcommand\Mn@minionint@bold{-Bold}
\DeclareOption{slides}{% 
  \def\Mn@minionint@opticals{-NoOpticals} 
  \PassOptionsToPackage{slides}{MinionPro-FontDef}}
\DeclareOption{noopticals}{% 
  \def\Mn@minionint@opticals{-NoOpticals} 
  \def\Mn@minionint@bold{-Bold} 
  \PassOptionsToPackage{noopticals}{MinionPro-FontDef}}
\PassOptionsToPackage{noopticals}{MinionPro-FontDef}
\DeclareOption{opticals}{% 
  \def\Mn@minionint@opticals{}% 
  \PassOptionsToPackage{opticals}{MinionPro-FontDef}}
\DeclareOption{smallfamily}{% 
  \def\Mn@minionint@bold{-Bold}% 
  \PassOptionsToPackage{smallfamily}{MinionPro-FontDef}}
\DeclareOption{medfamily}{% 
  \def\Mn@minionint@bold{-Semibold}% 
  \PassOptionsToPackage{medfamily}{MinionPro-FontDef}}
\DeclareOption{fullfamily}{% 
  \def\Mn@minionint@bold{-Semibold}% 
  \PassOptionsToPackage{fullfamily}{MinionPro-FontDef}}
\DeclareOption{normalsize}{% 
  \PassOptionsToPackage{normalsize}{MinionPro-FontDef}}
\DeclareOption{nonormalsize}{% 
  \PassOptionsToPackage{nonormalsize}{MinionPro-FontDef}}
\newcommand\Mn@Text@Fig{OsF}
\newcommand\Mn@Math@Fig{OsF}
\newcommand\Mn@Text@Family{MinionPro-\Mn@Text@Fig}
\newcommand\Mn@Math@Family{MinionPro-\Mn@Math@Fig}
\newcommand\Mn@Math@TFamily{MinionPro-T\Mn@Math@Fig}
\newcommand\Mn@Math@LetterShape{it}
\DeclareOption{textosf}{\def\Mn@Text@Fig{OsF}}
\DeclareOption{textlf} {
  \def\Mn@Text@Fig{LF}}
\DeclareOption{mathosf}{\def\Mn@Math@Fig{OsF}}
\DeclareOption{mathlf} {
  \def\Mn@Math@Fig{LF}}
\DeclareOption{osf}{\ExecuteOptions{textosf,mathosf}}
\DeclareOption{lf} {
  \ExecuteOptions{textlf,mathlf}}
\DeclareOption{mathtabular}{\let\Mn@Math@Family\Mn@Math@TFamily}
\newcommand\Mn@load@cal{} 
\newcommand\Mn@load@bb{} 
\newcommand\Mn@load@frak{} 

Calligraphic fonts

These hooks are executed once the math versions have been set up.
\newcommand\Mn@load@cal{}
\newcommand\Mn@load@bb{}
\newcommand\Mn@load@frak{}

Most options are handled by MnSymbol.
\PassOptionsToPackage{mnsy}{MnSymbol}
\PassOptionsToPackage{mnsy}{MnSymbol} 
\DeclareOption{mnsy}{
  \PassOptionsToPackage{mnsy}{MnSymbol}
  \def\Mn@load@cal{
    \SetMathAlphabet{\mathcal}{boldtabular}{\Mn@Symbol}{b}{n}
  }}
\DeclareOption{cnsy}{
  \PassOptionsToPackage{cnsy}{MnSymbol}
  \PassOptionsToPackage{cnsy}{MnSymbol}
}}
Greek letters
\Mn@greek@Upright, \Mn@greek@Mixed, and \Mn@greek@Italic are defined below in section 10.4 before \Mn@load@greek is executed.

We have to undefine \mathfrak and \mathbb before redefining them, because they might be defined in such a way that \DeclareMathAlphabet does not recognize them as math alphabets and refuses to overwrite their definitions (e.g., package eufm uses \newcommand{\mathfrak}{\EuFrak}).
\newcommand\Mn@load@fourierbb{
\let\mathbb\@undefined
\let\Bbbk\@undefined
\DeclareFontFamily{U}{futm}{}
\DeclareFontShape{U}{futm}{m}{n}{}{ <->s*[0.95] fourier-bb }
\DeclareMathAlphabet{\mathbb}{U}{futm}{m}{n}
\newcommand\Bbbk{\mathbb{k}}}
\DeclareOption{amsbb} {
\let\Mn@load@bb\Mn@load@amsbb}
\DeclareOption{lucidabb} {
\let\Mn@load@bb\Mn@load@lucidabb}
\DeclareOption{fourierbb} {
\let\Mn@load@bb\Mn@load@fourierbb}

Integrals
\newcommand\Mn@load@integrals{}
\DeclareOption{minionint} {
\def\Mn@load@integrals{\Mn@Decl@Minion@Ints}}

Miscellaneous options
Footnote figures, the g glyph in math mode, extra spacing for the apostrophe.
\DeclareOption{footnotefigures} {
\def\@makefnmark{\begingroup\normalfont
\fontfamily{MinionPro-Extra}\fontencoding{U}\selectfont
\@thefnmark\endgroup}}
\newcommand\Mn@Define@Open@g{}
\DeclareOption{openg} {
\def\Mn@Define@Open@g{\mathcode'g=8000
\DeclareMathSymbol{'g}{\mathalpha}{letters}{'g}
\begingroup
\lccode'~='g
\lowercase{\gdef~{\ifnum\the\mathgroup=\m@ne \openg \else \Mn@g \fi}}
\endgroup}}
\newcommand\Mn@Quote@Spacing{}
\DeclareOption{loosequotes} {
\def\Mn@Quote@Spacing{\Mn@Quote@Spacing@Loose}}

Defaults
\ExecuteOptions{amsbb,eufrak}
\ProcessOptions\relax

10.2 Font declarations
\RequirePackage{MinionPro-FontDef}
\@ifpackageloaded{textcomp}{}{\RequirePackage{textcomp}}
If no fraktur font is loaded then take the Euler font.

By default, we use \( b \) for the bold series. If MinionPro-Semibold is not available this might internally be mapped to MinionPro-Bold (see MinionPro-FontDef).

If a recent version of microtype is loaded then we implement an option to increase the side bearings of all quote glyphs.

Math fonts

Redefine the standard math versions normal and bold.
Extra math versions `tabular` and `boldtabular`, which use tabular figures instead of proportional ones. These math versions can be useful in tables (cf. section 2).

```latex
\DeclareMathVersion{tabular}
\SetSymbolFont{operators}{tabular} {T1} \{\Mn@Math@TFamily\} \{m\} \{n\}
\SetSymbolFont{letters} {tabular} {OML} \{MinionPro-TosF\} \{m\} \{\Mn@Math@LetterShape\}
\SetMathAlphabet\mathit {tabular} {T1} \{\Mn@Math@TFamily\} \{m\} \{it\}

\DeclareMathVersion{boldtabular}
\SetSymbolFont{operators}{boldtabular} {T1} \{\Mn@Math@TFamily\} \{eb\} \{n\}
\SetSymbolFont{letters} {boldtabular} {OML} \{MinionPro-TosF\} \{eb\} \{\Mn@Math@LetterShape\}
\SetMathAlphabet\mathit {boldtabular} {T1} \{\Mn@Math@TFamily\} \{eb\} \{it\}
```

Execute the hooks set up above to load the various math alphabets.

```latex
\Mn@load@bb
\Mn@load@frak
\Mn@load@cal
```

10.3 Font selection

The font selection commands such as `\figureversion`, `\textsw`, and `\textssc` are provided by the companion package `fontaxes`, which may be useful for other font families as well.

```latex
\RequirePackage{fontaxes}[2005/05/04]
```

We define an additional short hand for compatibility's sake.

```latex
\let\oldstylenums\textfigures
```

10.4 Greek letters

We provide math-mode commands for each Greek letter, both italic and upright. Furthermore, there are three commands to select the default version of the letters (all upright, all italic, or capitals upright and lowercase italic).

While declaring the Greek letters we collect the uppercase and lowercase letters in two lists. (We distinguish them by the first letter of their name.) These lists are then used to select the different versions.

```latex
\if@Mn@Math@
\newcommand\Mn@greek@list@upper{}
```

```latex
13
```
198 \newcommand\Mn@greek@list@lower{}
199 \let\Mn@greek@list@upper\@gobble
200 \let\Mn@greek@list@lower\@gobble

This macro holds one of the two list names.

201 \newcommand\Mn@greek@list{}
202 \newcommand*\Mn@greek@letter[3][]{%  
203 \expandafter\DeclareMathSymbol
204 \expandafter{\csname it#1\endcsname}{\mathord}{letters}{#2}%
205 \expandafter\DeclareMathSymbol
206 \expandafter{\csname up#1\endcsname}{\mathord}{letters}{#3}%
207 \edef\@tempa{'\@car#1\@nil}%
208 \edef\Mn@greek@list{\expandafter\noexpand\csname
209 Mn@greek@list@\ifnum\uccode\@tempa=\@tempa upper\else lower\fi\endcsname}%
210 \expandafter\edef\Mn@greek@list{\Mn@greek@list,#1}%
}

We can now declare the Greek letters (left italic, right upright).

212 \Mn@greek@letter{Gamma} {'000}{'200}
213 \Mn@greek@letter{Delta} {'001}{'201}
214 \Mn@greek@letter{Theta} {'002}{'202}
215 \Mn@greek@letter{Lambda} {'003}{'203}
216 \Mn@greek@letter{Xi} {'004}{'204}
217 \Mn@greek@letter{Pi} {'005}{'205}
218 \Mn@greek@letter{Sigma} {'006}{'206}
219 \Mn@greek@letter{Upsilon} {'007}{'207}
220 \Mn@greek@letter{Phi} {'010}{'210}
221 \Mn@greek@letter{Psi} {'011}{'211}
222 \Mn@greek@letter{Omega} {'012}{'212}
223 \Mn@greek@letter{alpha} {'013}{'213}
224 \Mn@greek@letter{beta} {'014}{'214}
225 \Mn@greek@letter{gamma} {'015}{'215}
226 \Mn@greek@letter{delta} {'016}{'216}
227 \Mn@greek@letter{epsilon} {'017}{'217}
228 \Mn@greek@letter{zeta} {'020}{'220}
229 \Mn@greek@letter{eta} {'021}{'221}
230 \Mn@greek@letter{theta} {'022}{'222}
231 \Mn@greek@letter{iota} {'023}{'223}
232 \Mn@greek@letter{kappa} {'024}{'224}
233 \Mn@greek@letter{lambda} {'025}{'225}
234 \Mn@greek@letter{mu} {'026}{'226}
235 \Mn@greek@letter{nu} {'027}{'227}
236 \Mn@greek@letter{xi} {'030}{'230}
237 \Mn@greek@letter{pi} {'031}{'231}
238 \Mn@greek@letter{rho} {'032}{'232}
239 \Mn@greek@letter{sigma} {'033}{'233}
240 \Mn@greek@letter{tau} {'034}{'234}
241 \Mn@greek@letter{upsilon} {'035}{'235}
242 \Mn@greek@letter{phi} {'036}{'236}
243 \Mn@greek@letter{chi} {'037}{'237}
244 \Mn@greek@letter{psi} {'040}{'240}
Some of the following symbols are not really Greek letters but are treated in the same way.

Go through a list #2 of Greek letters and \let them be their #1-prefixed variants.

Finally initialise the Greek letters.

10.5 pdfTeX to-unicode support

Old versions of MinionPro have non-standard glyph names.
\pdfglyphtounicode{uniEFF3}{02DA} \textsuperscript{\textdegree} ring.cap
\pdfglyphtounicode{uniEFF5}{02DC} \textsuperscript{\textdegree} tilde.cap
\pdfglyphtounicode{uniEFF7}{02C6} \textsuperscript{\textdegree} circumflex.cap
\pdfglyphtounicode{uniF628}{2030} \textperthousand.oldstyle
\pdfglyphtounicode{uniF62A}{0028} \textparenleft.denominator
\pdfglyphtounicode{uniF62C}{0029} \textparenright.denominator
\pdfglyphtounicode{uniF631}{0028} \textparenleft.numerator
\pdfglyphtounicode{uniF633}{0029} \textparenright.numerator
\pdfglyphtounicode{uniF638}{0030} \textzero.slash
\pdfglyphtounicode{uniF639}{0030} \textzero.fitted
\pdfglyphtounicode{uniF63A}{0032} \texttwo.fitted
\pdfglyphtounicode{uniF63B}{0033} \textthree.fitted
\pdfglyphtounicode{uniF63C}{0034} \textfour.fitted
\pdfglyphtounicode{uniF63D}{0035} \textfive.fitted
\pdfglyphtounicode{uniF63E}{0036} \textsix.fitted
\pdfglyphtounicode{uniF63F}{0037} \textseven.fitted
\pdfglyphtounicode{uniF640}{0038} \texteight.fitted
\pdfglyphtounicode{uniF641}{0039} \textnine.fitted
\pdfglyphtounicode{uniF642}{0025} \percent.oldstyle
\pdfglyphtounicode{uniF643}{0030} \textzero.taboldstyle
\pdfglyphtounicode{uniF644}{0031} \textone.taboldstyle
\pdfglyphtounicode{uniF645}{0032} \texttwo.taboldstyle
\pdfglyphtounicode{uniF646}{0033} \textthree.taboldstyle
\pdfglyphtounicode{uniF647}{0034} \textfour.taboldstyle
\pdfglyphtounicode{uniF648}{0035} \textfive.taboldstyle
\pdfglyphtounicode{uniF649}{0036} \textsix.taboldstyle
\pdfglyphtounicode{uniF64A}{0037} \textseven.taboldstyle
\pdfglyphtounicode{uniF64B}{0038} \texteight.taboldstyle
\pdfglyphtounicode{uniF64C}{0039} \textnine.taboldstyle
\pdfglyphtounicode{uniF64D}{20A1} \textcolonmonetary.taboldstyle
\pdfglyphtounicode{uniF64E}{20AC} \textEuro.taboldstyle
\pdfglyphtounicode{uniF64F}{0192} \textflorin.taboldstyle
\pdfglyphtounicode{uniF650}{0023} \textnumbersign.taboldstyle
\pdfglyphtounicode{uniF651}{0043} \textsterling.taboldstyle
\pdfglyphtounicode{uniF652}{00A5} \textyen.taboldstyle
\pdfglyphtounicode{uniF653}{0024} \textdollar.taboldstyle
\pdfglyphtounicode{uniF654}{00A2} \textcent.taboldstyle
\pdfglyphtounicode{uniF655}{0030} \textzero.denominator
\pdfglyphtounicode{uniF656}{0031} \textone.denominator
\pdfglyphtounicode{uniF657}{0032} \texttwo.denominator
\pdfglyphtounicode{uniF658}{0033} \textthree.denominator
\pdfglyphtounicode{uniF659}{0034} \textfour.denominator
\pdfglyphtounicode{uniF65A}{0035} \textfive.denominator
\pdfglyphtounicode{uniF65B}{0036} \textsix.denominator
\pdfglyphtounicode{uniF65C}{0037} \textseven.denominator
\pdfglyphtounicode{uniF65D}{0038} \texteight.denominator
\pdfglyphtounicode{uniF65E}{0039} \textnine.denominator
\pdfglyphtounicode{uniF65F}{002C} \textcomma.denominator
\pdfglyphtounicode{uniF660}{002E} \textperiod.denominator
\pdfglyphtounicode{uniF661}{0030} \textzero.numerator
10.6 Superior and inferior figures

We define commands to convert numbers to numerator figures and denominator figures.
\newcommand*{\Mn@extra@font}{\fontencoding{U}\fontfamily{MinionPro-Extra}\selectfont}
\newcommand*{\@numerator@fig}[1]{\@for@tok\@nf@fig:=#1\do{\ifcase\@nf@fig\char'00\or\char'01\or\char'02\or\char'03\or\char'04\or\char'05\or\char'06\or\char'07\or\char'10\or\char'11\else\@latex@error{invalid argument to \string\@numerator@fig}\fi}}
\newcommand*{\@denominator@fig}[1]{\@for@tok\@nf@fig:=#1\do{\ifcase\@nf@fig\char'20\or\char'21\or\char'22\or\char'23\or\char'24\else\@latex@error{invalid argument to \string\@denominator@fig}\fi}}
\newcommand*{\@superior@fig}[1]{\@for@tok\@nf@fig:=#1\do{\ifcase\@nf@fig\char'10\or\char'11\or\char'12\or\char'13\or\char'14\else\@latex@error{invalid argument to \string\@superior@fig}\fi}}
\newcommand*{\@inferior@fig}[1]{\@for@tok\@nf@fig:=#1\do{\ifcase\@nf@fig\char'20\or\char'21\or\char'22\or\char'23\or\char'24\else\@latex@error{invalid argument to \string\@inferior@fig}\fi}}
\ensur@text switches to text mode, if necessary.
\newcommand*\ensure@text[1]{%
\[ \frac{}{} \text{assembled numerical fractions.} \]

\newcommand\@smallfrac[2]{\leavevmode
  \setbox\@tempboxa
  \vbox{% 
    \baselineskip\z@skip% 
    \lineskip.25ex% 
    \lineskiplimit-\maxdimen 
    \ialign{\hfil##\hfil\crcr
      \vbox to 2.13ex{\vss\hbox{\@numerator@fig{#1}}\vskip.68ex}
      \leqvtop 1ex{\vbox{\hbox{\@denominator@fig{#2}}}}
    \noalign{\vskip-1.47ex}}}%
  \dp\@tempboxa=0.49ex%
  \box\@tempboxa}

\newcommand\@slantfrac[2]{\Mn@extra@font\@@numerator@fig{#1}\kern-0.05em/\kern-0.06em\@@denominator@fig{#2}}

\DeclareRobustCommand\smallfrac[2]{\ensure@text{\kern0.06em\@smallfrac{#1}{#2}\kern0.09em}}
\DeclareRobustCommand\slantfrac[2]{\ensure@text{\kern0.06em\@slantfrac{#1}{#2}\kern0.09em}}

\begin{verbatim}
10.7 Additional symbols
Some symbols missing from MnSymbol can be taken from MinionPro.
\if@Mn@Math@
  \let\hbar\undefined
  \DeclareMathSymbol{\hbar}{\mathord}{letters}{\char265}
  \DeclareMathSymbol{\uphbar}{\mathord}{letters}{\char300}
  \DeclareMathSymbol{\partial}{\mathord}{letters}{\char100}
  \DeclareMathSymbol{\uppartial}{\mathord}{letters}{\char340}
  \DeclareMathSymbol{\ell}{\mathord}{letters}{\char140}
  \DeclareMathSymbol{\upell}{\mathord}{letters}{\char380}
  \DeclareMathSymbol{\slashedzero}{\mathord}{letters}{\char257}
  \DeclareMathSymbol{\upimath}{\mathord}{letters}{\char373}
  \DeclareMathSymbol{\upjmath}{\mathord}{letters}{\char374}
  \ DeclareMathSymbol{\varsmallint}{\mathord}{letters}{\char376}
  \DeclareMathSymbol{\openg}{\mathalpha}{letters}{\char267}
  \DeclareRobustCommand{\lambdabar}{\middlebar\lambda}
  \DeclareRobustCommand{\lambdaslash}{\middleslash\lambda}
\fi
\end{verbatim}

Archaic Greek letters not provided by MinionPro.
\if@Mn@Text@
  \def\Qoppa{\reflectbox{p}}
  \def\Sampi{\begingroup\fontfamily{cmr}\fontencoding{LGR}\selectfont\char23\endgroup}
  \let\stigma\stigma
\fi
10.8 Integral symbols

We can also replace the integral signs from MnSymbol by those of MinionPro. The following definitions provide this as an option.

Replace MnSymbol by MnSymbolFI.

Make the original integral symbols available as \var....
Replace the symbols with the new integrals.

10.9 Open G support

We can replace the closed \( g \) with the open variant \( \partial \). The following definitions provide this as an option.

10.10 Logos

Correct logos.
\DeclareRobustCommand{\LaTeX}{L\kern-.32em% 
{\sbox\z@ T 
\vbox to\ht\z@{\hbox{\check@mathfonts 
\fontsize\sf@size\z@ 
\math@fontsfalse\selectfont 
A} 
\vss} 
\kern-.15em 
\TeX}
\fi 
\if@Mn@Text@ 
\normalfont 
\fi 
⟨/style⟩

### 10.11 AMS

Fix a bug in amsmath.sty which does not support math fonts without a skew char.

\def\macc@set@skewchar#1{%
\begingroup
\ifnum\mathgroup=\m@ne \let\@tempa\@ne
\else
\ifnum\skewchar\textfont\mathgroup=\m@ne \let\@tempa\@ne
\else \let\@tempa\mathgroup
\fi
\fi
\count@=\skewchar\textfont\@tempa
\ifnum\count@=\m@ne
\endgroup
\def\macc@skewchar{}
\else
\advance\count@='7100
\edef\@tempa{\endgroup
\mathchardef\noexpand\macc@skewchar=\number\count@\relax}%
\@tempa
\fi
\fi
\count@=\skewchar\textfont\@tempa
\ifnum\count@=\m@ne
\endgroup
\def\macc@skewchar{}
\else
\advance\count@='7100
\edef\@tempa{\endgroup
\mathchardef\noexpand\macc@skewchar=\number\count@\relax}%
\@tempa
\fi
\fi

#1%
}

Make the changes take effect. This concludes the main style file.

\if@Mn@Text@
\normalfont
\fi
⟨/style⟩

### 11 Support for character protrusion

The microtype configuration. All four MinionPro families use the same file (cf. section 12).

\SetProtrusion
[ name = MinionPro-OT1-Roman ]

23
{ encoding = OT1,
family = {MinionPro-OsF,MinionPro-LF,MinionPro-TOsF,MinionPro-TLF},
shape = n }
{
A = {40,40},
F = {60},
J = {90},
K = {50},
L = {60},
T = {50,50},
V = {40,40},
W = {30,30},
X = {50,50},
Y = {50,50},
k = {60},
r = {80},
t = {100},
v = {70,70},
w = {40,40},
x = {60,60},
y = {70,70},
i = {70,180},
} = {30,60},
[ = {100,160}, ] = {160,100},
\textendash = {390,480}, \textemdash = {220,270},
\textquotedblleft = {380,250}, \textquotedblright = {250,380},
\textquoteleft = {670,450}, \textquoteright = {450,670},
}

\SetProtrusion
[ name = MinionPro-T1-Roman,
load = MinionPro-OT1-Roman ]
{
encoding = T1,
family = {MinionPro-OsF,MinionPro-LF,MinionPro-TOsF,MinionPro-TLF},
shape = n }
{023 = {,40}, \% fft ligature
032 = {,50}, \% ft ligature
191 = {30,30}, \% Th ligature
127 = {620,700}, \% hyphen
\AE = {40}, \% AE
\quotesinglbase = {670,670}, \quotesinglebase = {370,370},
\guilsinglleft = {500,360}, \guilsingleright = {360,500},
\guillemotleft = {320,230}, \guillemotright = {230,320},
}
\SetProtrusion
[ name = MinionPro-OT1-Italic]
{ encoding = OT1,
  family = {MinionPro-OsF,MinionPro-LF,MinionPro-TOsF,MinionPro-TLF},
  shape = {it,sl,sw} }
{
  A = \{120,50\},
  B = \{90,-50\},
  C = \{50,-60\},
  D = \{70,-30\},
  E = \{90,-50\},
  F = \{100,-40\},
  G = \{50,-60\},
  H = \{70,-40\},
  I = \{150,-90\},
  J = \{250,-130\},
  K = \{80,-50\},
  L = \{90,60\},
  M = \{60,-40\},
  N = \{70,-40\},
  O = \{70,-30\},
  P = \{70,-110\},
  Q = \{40,-40\},
  R = \{80,-50\},
  S = \{70,-70\},
  T = \{130,\},
  U = \{70,-40\},
  V = \{120,30\},
  W = \{90,20\},
  X = \{50,\},
  Y = \{160,\},
  Z = \{50,-50\},
  d = \{60,-60\},
  f = \{,-190\},
  027 = \{,-70\}, \% ff ligature
  028 = \{,-60\}, \% fi ligature
  030 = \{,-30\}, \% ffi ligature
  j = \{-90,-150\},
  p = \{-40,\},
  r = \{,80\},
  t = \{,100\},
  v = \{90,\},
  w = \{60,10\},
  x = \{90,\},
  l = \{190,40\},
  ( = \{90,\}, ) = \{90,\},
  [ = \{90,90\}, ] = \{120,60\}
We have no protruding values for small caps yet. The following stubs are unnecessary at
the moment, but they are here as a reminder.

We have no protruding values for small caps yet. The following stubs are unnecessary at
the moment, but they are here as a reminder.
[ name = MinionPro-OT1-SmallcapsItalic ]
encoding = OT1,
family = {MinionPro-OsF,MinionPro-LF,MinionPro-TOsF,MinionPro-TLF},
shape = {scit,sscit} }

\SetProtrusion
[ name = MinionPro-T1-SmallcapsItalic,
load = MinionPro-OT1-SmallcapsItalic ]
encoding = T1,
family = {MinionPro-OsF,MinionPro-LF,MinionPro-TOsF,MinionPro-TLF},
shape = {scit,sscit} }

\SetProtrusion
[ name = MinionPro-other-Roman ]
encoding = {LGR,U,OT2,T2A,T2B,T2C,T5,X2},
family = {MinionPro-OsF,MinionPro-LF,MinionPro-TOsF,MinionPro-TLF},
shape = {n}
! = {70,180},
( = {60,30}, ) = {30,60},
[ = {100,160}, ] = {160,100},
{,} = {440,700},
. = {660,700},
: = {400,480},
; = {350,440},
- = {700,700},
\textendash = {390,480}, \textemdash = {220,270},
\textquotedblleft = {520,130}, \textquotedblright = {520,130},
\textquoteleft = {690,140}, \textquoteright = {470,230},
\textendash = {400,500}, \textemdash = {220,280},
\textquotedblleft = {380,250}, \textquotedblright = {250,380}

\SetProtrusion
[ name = MinionPro-other-Italic ]
encoding = {LGR,U,OT2,T2A,T2B,T2C,T5,X2},
family = {MinionPro-OsF,MinionPro-LF,MinionPro-TOsF,MinionPro-TLF},
shape = {it,sl,sw}
! = {190,40},
( = {90, }, ) = {90, },
[ = {90,90}, ] = {120,60},
{,} = {210,680},
. = {640,680},
: = {380,430},
; = {430},
- = {750,750},
\textquoteleft = {690,140}, \textquoteright = {470,230},
\textendash = {400,500}, \textemdash = {220,280},
\textquotedblleft = {520,130}, \textquotedblright = {520,130},

(/mtcfg)
12 Font definition files

As all the font definitions look the same we introduce macros to ease the configuration. These macros are stored in the file MinionPro-FontDef.sty which is included by every FD file. Note that MinionPro-FontDef.sty will be included several times and that we do not know in which context the code is executed. Therefore, we have to define all non-private commands as globals.

Since this package should be loadable in an FD file we have to avoid all \preambleonly commands. Therefore, we use \ProvidesFile instead of \ProvidesPackage.

We add a guard so that this file is executed only once even if it is included multiple times.

\begin{verbatim}
\ifx\Mn@DeclareFontShape\@undefined\else\endinput\fi
\end{verbatim}

We distinguish between being loaded directly or via \usepackage in the preamble by checking \@nodocument.

\begin{verbatim}
\ifx\@nodocument\relax\else\fi
\end{verbatim}

These are the default values if it is impossible to process options.

\begin{verbatim}
\newcommand\Mn@option@opticals{noopticals}
\newcommand\Mn@option@fontset{smallfamily}
\newdimen\Mn@option@normalsize
\global\Mn@option@normalsize10pt
\end{verbatim}

Whether we should adapt the configuration to the \normalsize of the document. This switch is only needed locally.

\begin{verbatim}
\ifx\@nodocument\relax\else\fi
\end{verbatim}

\begin{verbatim}
\DeclareOption{slides}{\let\Mn@option@opticals\CurrentOption}
\DeclareOption{opticals}{\let\Mn@option@opticals\CurrentOption}
\DeclareOption{noopticals}{\let\Mn@option@opticals\CurrentOption}
\DeclareOption{smallfamily}{\let\Mn@option@fontset\CurrentOption}
\DeclareOption{medfamily}{\let\Mn@option@fontset\CurrentOption}
\DeclareOption{fullfamily}{\let\Mn@option@fontset\CurrentOption}
\DeclareOption{normalsize}{\Mn@option@normalsizetrue}
\end{verbatim}

\begin{verbatim}
\ExecuteOptions{smallfamily,noopticals,normalsize}
\ProcessOptions\relax
\end{verbatim}

\begin{verbatim}
\end{verbatim}

\begin{verbatim}
\end{verbatim}
The method to determine the main font size is inspired by microtype's implementation.

We use \texttt{\otf@makeglobal} from \texttt{otfontdef} to "export" the definitions that are needed globally.

Configuration database

These commands help in setting up the configuration database. They do not need to be global. But the config database itself has to be.

#3 is added to all instances listed in #2 of configuration class #1. #3 is read with \texttt{nfss} catcodes.
Let us look at an example of how the configuration database looks internally for \((\text{shape}, \text{sw})\), which is specified below in three steps. The following lines show different depths of expansion of the macro \texttt{\textbackslash Mn@config@shape@sw}, which finally yields the complete configuration:

\texttt{\textbackslash Mn@config@shape@sw}
\texttt{\textbackslash Mn@config@xi \textbackslash Mn@config@xiv \textbackslash Mn@config@xv}
\texttt{<-8>otf*[spacing=l1]<-8.5>otf*[variant=swash]<-20>otf*MinionPro-It}

The following commands are used in the Declare...Family commands to access the previously built configuration database. They must be expandable. \#3 is used as a default if no entry is found in the database.

\texttt{\newcommand*{\textbackslash Mn@UseConfig}[2]{}}
\texttt{\newcommand*{\textbackslash Mn@UseConfigOrDefault}[3]{}}
\texttt{\newcommand*{\textbackslash Mn@TheConfig}[2]{}}
\texttt{\otf@makeglobal{\textbackslash Mn@UseConfig}}
\texttt{\otf@makeglobal{\textbackslash Mn@UseConfigOrDefault}}
\texttt{\otf@makeglobal{\textbackslash Mn@TheConfig}}

Here comes the configuration.

\texttt{\textbackslash Mn@AddToConfig\{opticals\}\{opticals\}\{<-8.5>otf* [optical=Capt]\}}
\texttt{\textbackslash Mn@AddToConfig\{<-8.5-13.1>otf* [optical=Text]\}}
\texttt{\textbackslash Mn@AddToConfig\{<13.1-20>otf* [optical=Subh]\}}
\texttt{\textbackslash Mn@AddToConfig\{<20->otf* [optical=Disp]\}}
\texttt{\textbackslash Mn@AddToConfig\{noopticals\}\{<-8.5>otf* [optical=Text]\}}
\texttt{\textbackslash Mn@AddToConfig\{slides\}\{<-8.5>otf* [optical=Capt]\}}
\texttt{\ifdim\textbackslash Mn@option@normalsize<10.1pt}}
\texttt{\textbackslash Mn@AddToConfig\{fontset/weight\}\{fullfamily/m\}\{<-6>otf* [weight=Semibold]\}}
\texttt{\textbackslash Mn@AddToConfig\{fontset/weight\}\{fullfamily/m\}\{<-6-8.5>otf* [weight=Medium]\}}
Substitutions

\Mn@AddToConfig{sub:series} {sb} {b}
\Mn@AddToConfig{sub:series} {bx} {b}
\Mn@AddToConfig{sub:shape} {sl} {it}
\Mn@AddToConfig{sub:shape} {scsl} {scit}
\Mn@AddToConfig{sub:shape} {sscsl} {sscit}
\Mn@AddToConfig{sub:shape} {scsw} {scit}
\Mn@AddToConfig{sub:shape} {sscsw} {sscit}
\Mn@AddToConfig{sub:encoding/shape}{TS1/sw}{it}

Code for the last argument of \DeclareFontShape

\Mn@AddToConfig{code:shape}{sw}{skewchar\font='337}

Declaration of font families and shapes

\newcommand*\MnDeclareFontShape[6]{}{%

Check if any substitutions are specified.
\edef\@tempa{%
\Mn@UseConfig{sub:series} {#4}%
\Mn@UseConfigOrDefault{sub:encoding/shape} {#2/#5}%
\Mn@UseConfig{sub:shape} {#5}%
\}
\ifx\@tempa\@empty
Collect the configuration and declare the font shape. \DeclareFontShape fully expands
its fifth argument (with our macros \Mn@UseConfig in it), but we have to retrieve the code
for the sixth argument ourselves.
\edef\@temppena{%
\DeclareFontShape{#2}{#3-#6}{#4}{#5}{%}
\Mn@UseConfig{opticals} {#4}%
\Mn@UseConfig{fontset/weight} {#4}%
\Mn@UseConfig{weight} {#4}%
\Mn@UseConfig{encoding/shape} {#2/#5}%
\Mn@UseConfig{shape} {#5}%
\}%
\edef\@tempa{\the\@temppena}\%
\else
Generate the substitution. (All substitutions are silent at the moment.)
\edef\@tempa{\the\@temppena}\%
\}
\endinput
An additional macro \csname\string\foo\endcsname is generated by \newcommand for processing an optional argument of \foo.

Adjust font dimension \texttt{#1} of the current font. The function in \texttt{#2} should replace the old value in dimen \texttt{\Mn@fontdimen} with a new one (which may depend on other parameters like \texttt{\f@size}).

We define font family aliases so that we can place all configurations for the MinionPro family variants into one \texttt{microtype} file: \texttt{mt-MinionPro.cfg}. We use \texttt{microtype}'s hook if \texttt{microtype} has not been loaded yet (which should be the case); otherwise we can execute the alias definitions directly.
Using these macros the various FD files become simple one-liners.

\input{MinionPro-FontDef.sty}

\DeclareSmallFontFamily{U}{MinionPro}
\DeclareSmallFontFamily{LGR}{MinionPro}
\DeclareSmallFontFamily{LGI}{MinionPro}
\DeclareLargeFontFamily{OT1}{MinionPro}
\DeclareLargeFontFamily{T1}{MinionPro}
\DeclareLargeFontFamily{LY1}{MinionPro}
\DeclareLargeFontFamily{T5}{MinionPro}
\DeclareSmallFontFamily{T2A}{MinionPro}
\DeclareSmallFontFamily{T2B}{MinionPro}
\DeclareSmallFontFamily{T2C}{MinionPro}
\DeclareSmallFontFamily{TS1}{MinionPro}
\DeclareSmallFontFamily{T5}{MinionPro}
\DeclareMathFontFamily{OML}{MinionPro}
\DeclareMathShape{m}{n}{it}{<-> ssub+MinionPro-TOSF/\Mn@series/\Mn@shape}