

Axes, axes, axes

Andreas Böhmann Michael Ummels

v1.0d – 2014/03/23

Abstract

The fontaxes package simulates multiple independent font selection axes on top of certain single NFSS axes: *base family*, *figure style*, and *figure alignment* on top of *family*; *primary shape* and *secondary shape* on top of *shape*; and *math weight* and *math figure alignment* on top of *math version*.

Contents

1	Introduction	1
2	Usage	1
2.1	Shape	2
2.2	Figure version	2
2.3	Math version	3
2.4	Additional commands	3
3	Implementation	4
3.1	High-level author commands (Level 1)	4
3.1.1	Shape	4
3.1.2	Figure version	5
3.1.3	Math version	6
3.2	Low-level author commands (Level 2)	6
3.3	Internals (Layer 3)	7
3.4	Encoding	9
3.5	Decoding	11
3.6	Compatibility	12
3.7	Tools	13
3.8	Tests	14

1 Introduction

The introduction of the New Font Selection Scheme (NFSS) has greatly simplified the usage of \LaTeX with fonts different from the Computer Modern fonts originally

designed for T_EX. However, the NFSS has some limitations. In particular, it defines only one axis for the font shape, which caters for both the actual *shape* of the font (e.g. upright, italic or slanted) and the *case* of the font (e.g. upper-lower case and small-caps). For example, if the current font shape is italic, then selecting small capitals using `\scshape` or `\textsc` will revert to an upright shape, even if the font has italic small capitals.

The `fontaxes` package alleviates the deficiencies of the NFSS by simulating multiple axes on top of single NFSS axes. In particular, it replaces the single NFSS shape axis by a primary and a secondary shape axis, catering for the shape and the case of the font, respectively. Moreover, the package introduces three new axes to deal with different *figure versions*, which are provided by many professional fonts.

2 Usage

You can load this package by adding

```
\usepackage{fontaxes}
```

to the preamble of your document. This redefines and makes available certain font selection commands, which are described in the rest of this section.

2.1 Shape

The `fontaxes` package splits the NFSS's single shape axis into two: the primary shape axis (`n`, `it`, etc.) and the secondary shape axis (`ulc`, `sc`, etc.).

`\upshape` The commands `\upshape`, `\itshape`, and `\slshape` are redefined to access the primary axis only. For access to a swash shape, the command `\swshape` has been added.

`\swshape` The commands `\scshape` and `\sscshape` (spaced small caps) access the secondary axis. To return from any small-caps shape to upper-lower case, you can use the command `\ulcshape`.

`\ulcshape` All these commands update the two shape axes using the low-level commands `\fontprimaryshape{<value>}` and `\fontsecondaryshape{<value>}`.

`\sscshape` If you want to change which values are used by the various commands `\<abbr>shape`, redefine the corresponding `\<abbr>default`. The additional commands `\swdefault`, `\sscdefault`, and `\ulcdefault` are provided with their default values `sw`, `ssc`, and `ulc`, respectively.

`\fontprimaryshape`
`\fontsecondaryshape`
`\swdefault`
`\sscdefault`
`\ulcdefault`

2.2 Figure version

Different figure versions are usually implemented as different font families (e.g. `MinionPro-0sF`, `LF`, `T0sF`, `TLF`) or `ppl{j,x}`). The `fontaxes` package splits off the axes *figure style* and *figure alignment*, which leaves the *base family* (e.g. `MinionPro` or `ppl`).

<code>\txfigures</code> <code>\lnfigures</code> <code>\tbfigures</code> <code>\prfigures</code> <code>\fontfigurestyle</code> <code>\fontfigurealignment</code> <code>\fontbasefamily</code>	<p>The fontaxes package knows two figure styles, text and lining (accessible via <code>\txfigures</code> and <code>\lnfigures</code>), and two modes of figure alignment, tabular and proportional (accessible via the switches <code>\tbfigures</code> and <code>\prfigures</code>).</p> <p>Additionally, you can access both axes directly using the low-level commands <code>\fontfigurestyle{<value>}</code> and <code>\fontfigurealignment{<value>}</code>.</p> <p>If you want to change the font family without changing the figure version, use <code>\fontbasefamily{<value>}</code>. (All <code>\font...</code> commands require a successive <code>\selectfont</code> to make the changes take effect.)</p> <p>For choosing the figure versions to be used in math mode, you can use the corresponding axis <i>math figure alignment</i>. Note that there is currently no means for changing the figure style used in math.</p>
--	---

2.3 Math version

<code>\boldmath</code> <code>\unboldmath</code> <code>\tabularmath</code> <code>\proportionalmath</code> <code>\mathweight</code> <code>\mathfigurealignment</code>	<p>By default, \TeX provides two math versions, normal and bold, as well as commands <code>\boldmath</code> and <code>\unboldmath</code> for switching between them. The fontaxes packages redefines these commands to operate on the axis <i>math weight</i>.</p> <p>A second axis <i>math figure alignment</i> is introduced that allows you to switch between tabular and proportional figures using <code>\tabularmath</code> and <code>\proportionalmath</code>. (This assumes the presence of additional math versions <code>tabular</code> and <code>boldtabular</code>; the package will copy the setups of math versions <code>normal</code> and <code>bold</code> at the end of the preamble in case you do not provide your own declarations.)</p> <p>You can directly assign values to the axes using the low-level commands <code>\mathweight{<value>}</code> and <code>\mathfigurealignment{<value>}</code>.</p> <p>Table 1 summarizes which commands set which values on which axes.</p>
--	--

2.4 Additional commands

<code>\textsw</code> <code>\textssc</code> <code>\textulc</code> <code>\textfigures</code> <code>\liningfigures</code> <code>\tabularfigures</code> <code>\proportionalfigures</code> <code>\figureversion</code>	<p>Similar to the well-known <code>\textit</code>, <code>\textsc</code>, etc. this package provides commands <code>\textsw</code>, <code>\textssc</code>, <code>\textulc</code>, <code>\textfigures</code>, <code>\liningfigures</code>, <code>\tabularfigures</code> and <code>\proportionalfigures</code> that take one argument and apply the font change only to the argument. For example, <code>\textsw{<text>}</code> is roughly equivalent to <code>{\swshape<text>}</code> (but automatically adds italic corrections).</p> <p>The command <code>\figureversion{<options>}</code> allows easy switching of multiple aspects of figures simultaneously. It takes as an argument a comma-separated list of one or more of the following options:</p>
--	---

<code>text, osf</code>	for text figures,
<code>lining, lf</code>	for lining figures,
<code>tabular, tab</code>	for tabular figures,
<code>proportional, prop</code>	for proportional figures.

For example, `\figureversion{lf, tab}` selects tabular lining figures.

Table 1: Summary of commands

Command	Axis	Value	Default
<code>\upshape</code>	<code>\fontprimaryshape</code>	<code>\updefault</code>	n
<code>\itshape</code>		<code>\itdefault</code>	it
<code>\slshape</code>		<code>\sldefault</code>	sl
<code>\swshape</code>		<code>\swdefault</code>	sw
<code>\ulcshape</code>	<code>\fontsecondaryshape</code>	<code>\ulcdefault</code>	ulc
<code>\scshape</code>		<code>\scdefault</code>	sc
<code>\sscshape</code>		<code>\sscdefault</code>	ssc
<code>\txfigures</code>	<code>\fontfigurestyle</code>	text	
<code>\lnfigures</code>		lining	
<code>\tbfigures</code>	<code>\fontfigurealignment</code>	tabular	
<code>\prfigures</code>		proportional	
<code>\langle none \rangle</code>	<code>\fontbasefamily</code>	<i>\langle font-dependent \rangle</i>	
<code>\boldmath</code>	<code>\mathweight</code>	bold	
<code>\unboldmath</code>		normal	
<code>\tabularmath</code>	<code>\mathfigurealignment</code>	tabular	
<code>\proportionalmath</code>		proportional	

3 Implementation

3.1 High-level author commands (Level 1)

3.1.1 Shape

```

\upshape Axis 1: primary shape
\itshape 1 \(*package)
\slshape 2 \DeclareRobustCommand\upshape{\not@math@alphabet\upshape\relax
\swshape 3 \fontprimaryshape\updefault\selectfont}
4 \DeclareRobustCommand\itshape{\not@math@alphabet\itshape\mathit
5 \fontprimaryshape\itdefault\selectfont}
6 \DeclareRobustCommand\slshape{\not@math@alphabet\slshape\relax
7 \fontprimaryshape\sldefault\selectfont}
8 \DeclareRobustCommand\swshape{\not@math@alphabet\swshape\relax
9 \fontprimaryshape\swdefault\selectfont}

\scshape Axis 2: secondary shape
\sscshape 10 \DeclareRobustCommand\scshape{\not@math@alphabet\scshape\relax
\ulcshape 11 \fontsecondaryshape\scdefault\selectfont}
12 \DeclareRobustCommand\sscshape{\not@math@alphabet\sscshape\relax
13 \fontsecondaryshape\sscdefault\selectfont}
14 \DeclareRobustCommand\ulcshape{\not@math@alphabet\ulcshape\relax
15 \fontsecondaryshape\ulcdefault\selectfont}

```

`\noscsshape` Provide an alias for compatibility with the `slantsc` package.
 16 `\let\noscsshape\ulcshape`

`\swdefault`
`\ulcdefault` 17 `\providecommand\swdefault{sw}`
`\sscdefault` 18 `\providecommand\ulcdefault{ulc}`
 19 `\providecommand\sscdefault{ssc}`

`\textsw`
`\textssc` 20 `\DeclareTextFontCommand{\textsw}{\swshape}`
`\textulc` 21 `\DeclareTextFontCommand{\textssc}{\sscshape}`
 22 `\DeclareTextFontCommand{\textulc}{\ulcshape}`

3.1.2 Figure version

`\txfigures` Axis 1: figure style
`\lnfigures` 23 `\def\txfigures{\@nomath\txfigures`
 24 `\fontfigurestyle{text}\selectfont}`
 25 `\def\lnfigures{\@nomath\lnfigures`
 26 `\fontfigurestyle{lining}\selectfont}`

`\tbfigures` Axis 2: figure alignment
`\prfigures` 27 `\def\tbfigures{\@nomath\tbfigures`
 28 `\fontfigurealignment{tabular}\selectfont}`
 29 `\def\prfigures{\@nomath\prfigures`
 30 `\fontfigurealignment{proportional}\selectfont}`

`\figureversion` This code originally appeared in the package `MinionPro`. We have adapted it to work within `fontaxes`' framework and also changed some option names.

```
31 \newcommand\fontaxes@fv@prefix{fontaxes@fv@switch@}
32 \newcommand*\fontaxes@fv@newoption[1]%
33   {\expandafter\newcommand\csname\fontaxes@fv@prefix #1\endcsname}
34 \fontaxes@fv@newoption{text}{\txfigures}
35 \fontaxes@fv@newoption{osf}{\txfigures}
36 \fontaxes@fv@newoption{lining}{\lnfigures}
37 \fontaxes@fv@newoption{lf}{\lnfigures}
38 \fontaxes@fv@newoption{tabular}{\tbfigures\tabularmath}
39 \fontaxes@fv@newoption{tab}{\tbfigures\tabularmath}
40 \fontaxes@fv@newoption{proportional}{\prfigures\proportionalmath}
41 \fontaxes@fv@newoption{prop}{\prfigures\proportionalmath}
```

We simply iterate over the list of figure versions specified in the argument to `\figureversion` and check if we have specified a matching option.

```
42 \newcommand\fontaxes@fv@list{}
43 \newcommand\fontaxes@fv{}
44 \DeclareRobustCommand*\figureversion[1]{%
45   \edef\fontaxes@fv@list{\zap@space#1 \@empty}%
46   \@for\fontaxes@fv:=\fontaxes@fv@list\do{%
47     \ifundefined{\fontaxes@fv@prefix\fontaxes@fv}{%
```

```

48     \PackageWarning{fontaxes}%
49     {Unknown figure style '\fontaxes@fv'\MessageBreak
50      specified as the argument to \string\figureversion.\MessageBreak
51      Figure style not changed}%
52   }{%
53     \nameuse{\fontaxes@fv@prefix\fontaxes@fv}%
54   }%
55 }%
56 }

```

Axis 3: base family `\fontbasefamily{...}`

```

\textfigures
\liningfigures 57 \DeclareTextFontCommand{\textfigures}{\txfigures}
\tabularfigures 58 \DeclareTextFontCommand{\liningfigures}{\lnfigures}
\proportionalfigures 59 \DeclareTextFontCommand{\tabularfigures}{\tbfigures\tabularmath}
60 \DeclareTextFontCommand{\proportionalfigures}
61   {\prfigures\proportionalmath}

```

3.1.3 Math version

```

\boldmath Axis 1: weight
\unboldmath 62 \def\boldmath{\@nomath\boldmath
63   \mathweight{bold}}
64 \def\unboldmath{\@nomath\unboldmath
65   \mathweight{normal}}

```

```

\tabularmath Axis 2: figure alignment
\proportionalmath 66 \def\tabularmath{\@nomath\tabularmath
67   \mathfigurealignment{tabular}}
68 \def\proportionalmath{\@nomath\proportionalmath
69   \mathfigurealignment{proportional}}

```

3.2 Low-level author commands (Level 2)

```

\mathweight{bold,normal} sets \mathversion;
\mathfigurealignment{tabular,proportional} sets \mathversion;
\fontfigurestyle{text,lining} sets \fontfamily;
\fontfigurealignment{tabular,proportional} sets \fontfamily;
\fontbasefamily{...} sets \fontfamily;
\fontprimaryshape{n,it,sl,sw} sets \fontshape;
\fontsecondaryshape{ulc,sc,ssc} sets \fontshape.

```

```

\mathweight
\mathfigurealignment 70 \DeclareRobustCommand\mathweight[1]{%
71   \fontaxes@get@math\edef\fontaxes@math@weight{#1}\fontaxes@set@math}
72 \DeclareRobustCommand\mathfigurealignment[1]{%
73   \fontaxes@get@math\edef\fontaxes@math@align{#1}\fontaxes@set@math}

```

```

\fontfigurestyle
\fontfigurealignment
\fontbasefamily
74 \DeclareRobustCommand\fontfigurestyle[1]{%
75 \fontaxes@get@family\edef\fontaxes@figure@style{#1}\fontaxes@set@family}
76 \DeclareRobustCommand\fontfigurealignment[1]{%
77 \fontaxes@get@family\edef\fontaxes@figure@align{#1}\fontaxes@set@family}
78 \DeclareRobustCommand\fontbasefamily[1]{%
79 \fontaxes@get@family\edef\fontaxes@family@base{#1}\fontaxes@set@family}

\fontprimaryshape
\fontsecondaryshape
80 \DeclareRobustCommand\fontprimaryshape[1]{%
81 \fontaxes@get@shape\edef\fontaxes@shape@one{#1}\fontaxes@set@shape}
82 \DeclareRobustCommand\fontsecondaryshape[1]{%
83 \fontaxes@get@shape\edef\fontaxes@shape@two{#1}\fontaxes@set@shape}

```

We have made most commands robust to protect them in moving arguments (e.g. section titles). Additionally, we want these commands to be ignored when hyperref is building PDF strings (e.g. for bookmarks).

```

84 \AtBeginDocument{
85 \ifpackageloaded{hyperref}{%
86 \pdfstringdefDisableCommands{%
87 \let\fontprimaryshape\@gobble
88 \let\fontsecondaryshape\@gobble
89 \let\fontfigurestyle\@gobble
90 \let\fontfigurealignment\@gobble
91 \let\fontbasefamily\@gobble
92 \let\textfigures\@firstofone
93 \let\liningfigures\@firstofone
94 \let\tabularfigures\@firstofone
95 \let\proportionalfigures\@firstofone
96 \let\textsw\@firstofone
97 \let\textssc\@firstofone
98 \let\textulc\@firstofone
99 }%
100 }{}%
101 }

```

3.3 Internals (Layer 3)

```

\fontaxes@set@math sets \mathversion;
\fontaxes@set@family sets \fontfamily;
\fontaxes@set@shape sets \fontshape.

```

```

\fontaxes@math@weight
\fontaxes@math@align
\fontaxes@family@base
\fontaxes@figure@style
\fontaxes@figure@align
\fontaxes@shape@one
\fontaxes@shape@two
The macros that hold the current values of the axes (here with some default values that will most certainly be overwritten during initialization; see \fontaxes@get@...).
102 \newcommand*\fontaxes@math@weight{normal}
103 \newcommand*\fontaxes@math@align{proportional}
104 \newcommand*\fontaxes@family@base{cmr}

```

```

105 \newcommand*\fontaxes@figure@style{lining}
106 \newcommand*\fontaxes@figure@align{proportional}
107 \newcommand*\fontaxes@shape@one{n}
108 \newcommand*\fontaxes@shape@two{ulc}

```

```

\fontaxes@set@math
\fontaxes@set@family 109 \newcommand*\fontaxes@set@math{%
\fontaxes@set@shape 110 \fontaxes@encode@math
111 \mathversion{\fontaxes@code}%
112 \fontaxes@save\math@version}
113 \newcommand*\fontaxes@set@family{%
114 \fontaxes@encode@family
115 \fontfamily{\fontaxes@code}%
116 \fontaxes@save\f@family}
117 \newcommand*\fontaxes@set@shape{%
118 \fontaxes@encode@shape
119 \fontshape{\fontaxes@code}%
120 \fontaxes@save\f@shape}

```

\fontaxes@get@math Check for changes: if changed, try to decode and update axes.

```

\fontaxes@get@family 121 \newcommand*\fontaxes@get@math{%
\fontaxes@get@shape 122 \iffontaxes@changed\math@version{%
123 \fontaxes@decode@{math}{\math@version}%
124 \ifx\fontaxes@edoc\relax\else
125 \edef\fontaxes@math@weight{\expandafter\@firstoftwo\fontaxes@edoc}%
126 \edef\fontaxes@math@align{\expandafter\@secondoftwo\fontaxes@edoc}%
127 \fi
128 \fontaxes@save\math@version
129 }{}}%
130 }

```

```

131 \newcommand*\fontaxes@get@family{%
132 \iffontaxes@changed\f@family{%
133 \let\fontaxes@edoc\relax
134 \expandafter\fontaxes@split@family\f@family--\@nnil
135 \ifx\fontaxes@split@suffix\relax\else
136 \fontaxes@decode@{figures}{\fontaxes@split@suffix}%
137 \fi
138 \ifx\fontaxes@edoc\relax

```

Try alternative.

```

139 \expandafter\fontaxes@split@familyalt\f@family
140 \@empty\@empty\@empty\@empty\@nnil
141 \ifx\fontaxes@split@suffix\relax\else
142 \fontaxes@decode@{figuresalt}{\fontaxes@split@suffix}%
143 \fi
144 \ifx\fontaxes@edoc\relax
145 \fontaxes@warn@undecodable{family '\f@family'}%
146 \edef\fontaxes@family@base{\f@family}%
147 \else

```



```

148     \edef\fontaxes@family@base{\fontaxes@split@prefix}%
149     \edef\fontaxes@figure@style{\expandafter\@firstoftwo\fontaxes@edoc}%

```

Do not overwrite align (does not occur in alternative naming scheme).

```

150     \fi
151     \else

```

Store values.

```

152     \edef\fontaxes@family@base{\fontaxes@split@prefix}%
153     \edef\fontaxes@figure@style{\expandafter\@firstoftwo\fontaxes@edoc}%
154     \edef\fontaxes@figure@align{\expandafter\@secondoftwo\fontaxes@edoc}%
155     \fi
156   }{}%
157 }

```

```

158 \newcommand*\fontaxes@get@shape{%
159   \iffontaxes@changed\f@shape{%
160     \fontaxes@decode@{shape}{\f@shape}%
161     \ifx\fontaxes@edoc\relax\else
162       \edef\fontaxes@shape@one{\expandafter\@firstoftwo\fontaxes@edoc}%
163       \edef\fontaxes@shape@two{\expandafter\@secondoftwo\fontaxes@edoc}%
164       \fi
165       \fontaxes@save\f@shape
166     }{}%
167 }

```

3.4 Encoding

```

\fontaxes@encode@math
\fontaxes@encode@family 168 \newcommand*\fontaxes@encode@math{%
\fontaxes@encode@figures 169 \fontaxes@encode@{math}{\fontaxes@math@weight}{\fontaxes@math@align}}%
\fontaxes@encode@figuresalt 170 }
\fontaxes@encode@shape

```

Default is concatenation.

```

171 \newcommand*\fontaxes@encode@math@default{%
172   \edef\fontaxes@code{\fontaxes@math@weight\fontaxes@math@align}}
173 \newcommand*\fontaxes@encode@family{%
174   \fontaxes@encode@{family}
175   {\fontaxes@family@base}{\fontaxes@figure@style}{\fontaxes@figure@align}}%
176 }

```

Try different naming conventions.

```

177 \newcommand*\fontaxes@encode@family@default{%
178   \fontaxes@encode@figures
179   \edef\fontaxes@code{\fontaxes@family@base-\fontaxes@code}%
180   \fontaxes@check@family\fontaxes@code
181   \iffontaxes@exists\else
182     \fontaxes@encode@figuresalt
183     \edef\fontaxes@code{\fontaxes@family@base\fontaxes@code}%
184     \fontaxes@check@family\fontaxes@code
185     \iffontaxes@exists\else

```

```

186     \edef\fontaxes@code{\fontaxes@family@base}%
187     \fi
188     \fi
189 }

190 \newcommand*\fontaxes@encode@figures{%
191   \fontaxes@encode@{figures}{\fontaxes@figure@style}{\fontaxes@figure@align}}%
192 }
193 \newcommand*\fontaxes@encode@figures@default{%
194   \edef\fontaxes@code{0sF}%
195   \PackageWarning{fontaxes}{Unknown figure version
196     '\fontaxes@figure@style\space + \fontaxes@figure@align'\MessageBreak
197     Encoding to '\fontaxes@code'}%
198 }

199 \newcommand*\fontaxes@encode@figuresalt{%
200   \fontaxes@encode@{figuresalt}{\fontaxes@figure@style}{\fontaxes@figure@align}}%
201 }
202 \newcommand*\fontaxes@encode@figuresalt@default{%
203   \PackageWarning{fontaxes}{Unknown figure version
204     '\fontaxes@figure@style\space + \fontaxes@figure@align'\MessageBreak
205     Encoding to '\fontaxes@code'}%
206   \edef\fontaxes@code{j}%
207 }

208 \newcommand*\fontaxes@encode@shape{%
209   \fontaxes@encode@[shape]{\fontaxes@shape@one}{\fontaxes@shape@two}}%
210 }

```

Default is (reverse) concatenation.

```

211 \newcommand*\fontaxes@encode@shape@default{%
212   \edef\fontaxes@code{\fontaxes@shape@two\fontaxes@shape@one}%
213 }

```

\fontaxes@encode@

```

214 \newcommand*\fontaxes@encode@[2]{%
215   \@ifundefined{fontaxes@encode@#1#2}
216     {\@nameuse{fontaxes@encode@#1@default}}
217     {\edef\fontaxes@code{\@nameuse{fontaxes@encode@#1#2}}}%
218 }

```

\fontaxes@naming@exception To do: Add a user interface to specify naming exceptions.

```

219 \newcommand*\fontaxes@naming@exception[3]{%
220   \expandafter\edef\curname fontaxes@encode@#1#2\endcsname{#3}%
221 }

```

The following alias is defined for compatibility with package files generated by autoinst.

```

222 \let\fa@naming@exception\fontaxes@naming@exception

```

The defaults n and ulc disappear when combined.

```

223 \fontaxes@naming@exception{shape}{n}{ulc}{n}

```

```

224 \fontaxes@naming@exception{shape}{\n}{sc}{sc}
225 \fontaxes@naming@exception{shape}{\n}{ssc}{ssc}
226 \fontaxes@naming@exception{shape}{\it}{ulc}{it}
227 \fontaxes@naming@exception{shape}{\sl}{ulc}{sl}
228 \fontaxes@naming@exception{shape}{\sw}{ulc}{sw}

```

The defaults disappear in the concatenation. `boldtabular` is formed regularly.

```

229 \fontaxes@naming@exception{math}{\normal}{proportional}{normal}
230 \fontaxes@naming@exception{math}{\normal}{tabular}{tabular}
231 \fontaxes@naming@exception{math}{\bold}{proportional}{bold}

```

Provide abbreviations for font family suffixes.

```

232 \fontaxes@naming@exception{figures}{\text}{proportional}{OsF}
233 \fontaxes@naming@exception{figures}{\text}{tabular}{TOsF}
234 \fontaxes@naming@exception{figures}{\lining}{proportional}{LF}
235 \fontaxes@naming@exception{figures}{\lining}{tabular}{TLF}

```

The `j/x` naming convention does not know about different figure alignments; let us silently ignore these.

```

236 \fontaxes@naming@exception{figuresalt}{\text}{proportional}{j}
237 \fontaxes@naming@exception{figuresalt}{\text}{tabular}{j}
238 \fontaxes@naming@exception{figuresalt}{\lining}{proportional}{x}
239 \fontaxes@naming@exception{figuresalt}{\lining}{tabular}{x}

```

3.5 Decoding

Detect if `\mathversion`, `\fontshape`, `\fontfamily` have been used not under control of this package.

```

\fontaxes@figure@style@domain Assuming an injective encoding function, we can construct decoding tables when
\fontaxes@figure@align@domain we know the function's domain. To do: Warn if decoding entries are overwritten
  \fontaxes@shape@one@domain (if the function is not injective).
  \fontaxes@shape@two@domain
\fontaxes@math@weight@domain
  \fontaxes@math@align@domain

```

```

240 \newcommand*\fontaxes@figure@style@domain{text,lining}
241 \newcommand*\fontaxes@figure@align@domain{proportional,tabular}
242 \newcommand*\fontaxes@shape@one@domain{n,it,sl,sw}
243 \newcommand*\fontaxes@shape@two@domain{ulc,sc,ssc}
244 \newcommand*\fontaxes@math@weight@domain{normal,bold}
245 \newcommand*\fontaxes@math@align@domain{proportional,tabular}

```

```

\fontaxes@create@decode@table #1 name, #2 list of axes

```

```

246 \newcommand*\fontaxes@create@decode@table[2]{%
247   \begingroup
248   \fontaxes@foreach{#2}{%
249     \@nameuse{fontaxes@encode@#1}%
250     \global\expandafter
251     \edef\csname fontaxes@decode@#1{\fontaxes@code}\endcsname{#2}%
252   }%
253   \endgroup
254 }

```

```

255 \AtEndOfPackage{
256   \fontaxes@create@decode@table{figures}
257     {\fontaxes@figure@style}{\fontaxes@figure@align}}
258   \fontaxes@create@decode@table{figuresalt}
259     {\fontaxes@figure@style}{\fontaxes@figure@align}}
260   \fontaxes@create@decode@table{shape}
261     {\fontaxes@shape@one}{\fontaxes@shape@two}}
262   \fontaxes@create@decode@table{math}
263     {\fontaxes@math@weight}{\fontaxes@math@align}}
264 }

```

`\fontaxes@warn@undecodable`

```

265 \newcommand*\fontaxes@warn@undecodable[1]{%
266   \PackageWarning{fontaxes}{I don't know how to decode\MessageBreak #1}}

```

`\fontaxes@decode@` Interpret the decoding tables.

```

267 \newcommand*\fontaxes@decode@[2]{%
268   \@ifundefined{fontaxes@decode@#1{#2}}{%
269     \let\fontaxes@edoc\relax
270     \fontaxes@warn@undecodable{#1 '#2'}%
271   }\edef\fontaxes@edoc{\@nameuse{fontaxes@decode@#1{#2}}}%
272 }

```

`\fontaxes@save` Save states of macros for future comparison.
`\iffontaxes@changed`

```

273 \newcommand*\iffontaxes@changed[1]{%
274   \expandafter\ifx\csname fontaxes@last@\string#1\endcsname#1%
275   \expandafter\@secondoftwo
276   \else
277     \expandafter\@firstoftwo
278   \fi
279 }
280 \newcommand*\fontaxes@save[1]{%
281   \expandafter\let\csname fontaxes@last@\string#1\endcsname#1%
282 }

```

3.6 Compatibility

`\fontaxes@provide@mv@copy` Declare math version #1 to be a copy of math version #2 if #1 does not exist already. To accomplish this, we have to know that a math version's configuration is basically stored in a macro `\mv@<name>` (which makes us dependent on the NFSS implementation; sigh ...).

```

283 \newcommand*\fontaxes@provide@mv@copy[2]{%
284   \@ifundefined{mv@#1}{%
285     \DeclareMathVersion{#1}%
286     \expandafter\let\csname mv@#1\expandafter\endcsname
287       \csname mv@#2\endcsname
288   }{}%
289 }

```

If no math versions `tabular` and `boldtabular` are defined in the preamble, we provide defaults by copying the states of `normal` and `bold` (assuming, in turn, that these two exist).

```
290 \AtBeginDocument{%
291   \fontaxes@provide@mv@copy{tabular}{normal}%
292   \fontaxes@provide@mv@copy{boldtabular}{bold}%
293 }
```

3.7 Tools

`\fontaxes@check@family` Check if family switching would yield an existing shape.

```
\iffontaxes@exists
294 \newif\iffontaxes@exists
295 \newcommand*\fontaxes@check@family[1]{%
296   \begingroup
297   \fontfamily{#1}\try@load@fontshape
298   \expandafter
299   \ifx\csname\curr@fontshape\endcsname\relax
300     \aftergroup\fontaxes@existsfalse
301   \else
302     \aftergroup\fontaxes@existstrue
303   \fi
304 \endgroup
305 }
```

`\fontaxes@split@prefix` The results of splitting a family name.

```
\fontaxes@split@suffi
306 \newcommand*\fontaxes@split@prefix{}
307 \newcommand*\fontaxes@split@suffi{}
```

`\fontaxes@split@family` Font name contains one hyphen; split there.

```
308 \newcommand*\fontaxes@split@family{}
309 \def\fontaxes@split@family#1-#2-#3\@nnil{%
310   \let\fontaxes@split@prefix\relax
311   \let\fontaxes@split@suffi\relax
312   \def\@tempa{#3}%
313   \ifx\@tempa\@empty\else
314     \def\fontaxes@split@suffi{#2}%
315     \ifx\fontaxes@split@suffi\@empty
316       \let\fontaxes@split@suffi\relax
317     \else
318       \def\fontaxes@split@prefix{#1}%
319     \fi
320   \fi
321 }
```

`\fontaxes@split@familyalt` Name consists of four characters; split off the last one. If there are just three characters, the default suffix is ‘x’.

```
322 \newcommand*\fontaxes@split@familyalt{}
323 \def\fontaxes@split@familyalt#1#2#3#4#5\@nnil{%
```

```

324 \let\fontaxes@split@prefix\relax
325 \let\fontaxes@split@suffix\relax
326 \edef\@tempa{#5}%
327 \ifx\@tempa\@empty
328   \ifx\@empty#4%
329     \def\fontaxes@split@prefix{#1#2#3}%
330     \def\fontaxes@split@suffix{x}%
331   \else
332     \def\fontaxes@split@prefix{#1#2#3}%
333     \def\fontaxes@split@suffix{#4}%
334   \fi
335 \fi
336 }

```

`\fontaxes@foreach` Execute #2 for each combination of values of the axes given in #1 (in the form `{\cs}{\cs}...`).

```

337 \newcommand\fontaxes@foreach[2]{%
338   \begingroup
339   \def\fontaxes@foreach@{#2}%
340   \@tfor\@tempa:=#1\do{%
341     \@temptokena\expandafter{\fontaxes@foreach@}%
342     \edef\fontaxes@foreach@{%
343       \noexpand\@for
344       \expandafter\noexpand\@tempa:=%
345       \expandafter\noexpand\csname
346       \expandafter\expandafter
347       \expandafter\@gobble
348       \expandafter\string\@tempa
349       @domain%
350     \endcsname
351     \noexpand\do{\the\@temptokena}%
352   }%
353 }%
354 \expandafter\endgroup\fontaxes@foreach@
355 }
356 \end{package}

```

3.8 Tests

The file `test-fontaxes.tex` (docstrip target `test`) exercises some features of `fontaxes`. Since it is rather ad-hoc code, it is not shown here. (It also requires the `MinionPro` package.)