

# The xltextra package

Will Robertson

2010/05/27     v0.5c

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Usage	2
<b>2</b>	<b>Features</b>	<b>2</b>
2.1	<code>\textsuperscript</code> and <code>\textsubscript</code>	2
2.2	Inner emphasis	3
2.3	Unicode footnote symbols	3
2.4	Verbatim	3
2.5	Logos	4
2.6	Discretionary hyphenation: <code>\-</code>	4
2.7	Vulgar fractions	4
2.8	Named glyphs	5
2.9	The <code>\showhyphens</code> command	5
<b>I</b>	<b>The xltextra package</b>	<b>6</b>
<b>3</b>	<b>Programmming bits and pieces</b>	<b>7</b>
<b>4</b>	<b>Logos</b>	<b>7</b>
4.1	Unicode footnote symbols	8
4.2	Emph	8
4.3	<code>\-</code>	8
4.4	Subscript and superscript	8
4.5	Verbatims	12

# 1 Introduction

This document describes the `xltextra` package. It implements some odds-and-ends features and improved functionality for broken or sub-standard L<sup>A</sup>T<sub>E</sub>X methods when using the X<sub>Y</sub>T<sub>E</sub>X format.

## 1.1 Usage

Easy: `\usepackage{xltextra}`. This package automatically loads the following packages: `fixltx2e`, `metalogo`, `xunicode`, `fontspec`.

There are some package options to disable various functionality that could clash with other things:

**no-sscript** Swaps the definitions of `\textsubscript` and `\textsuperscript` with their respective starred versions, as described in section §2.1.

**no-emph** Disables the redefinition of `\emph` and `\em` described in section §2.2.

**no-logos** Disables the redefinition of `\TeX`, etc. described in section §2.5, but *does* still define the `\XeTeX` and `\XeLaTeX` logo commands.

**no-hyphen** Disables the redefinition of `\-` (probably harmless anyway) described in section §2.6.

**no-verb** Disables the redefinition of `\verb*` and `\begin{verbatim}`, and the patching of various verbatim packages, as described in section §2.4.

# 2 Features

## 2.1 `\textsuperscript` and `\textsubscript`

These two macros have been redefined to take advantage, if possible, of actual superior or inferior glyphs in the main document font. This is very important for high-quality typesetting — compare this first example to the third; yes, they are the same font.

<code>\textsuperscript</code>	<code>abcdefghijklmnopqrstuvwxyz1234567890</code>
<code>\textsubscript</code>	<code>abcdefghijklmnopqrstuvwxyz1234567890</code>

But will fall back on ‘faked’ ones if they don’t exist: (this is Didot)

<code>\textsuperscript</code>	<code>abcdefghijklmnopqrstuvwxyz1234567890</code>
<code>\textsubscript</code>	<code>abcdefghijklmnopqrstuvwxyz1234567890</code>

The original definitions are available in starred versions of the commands:

<code>\textsuperscript*</code>	<code>abcdefghijklmnopqrstuvwxyz1234567890</code>
<code>\textsubscript*</code>	<code>abcdefghijklmnopqrstuvwxyz1234567890</code>

But beware fonts lacking the full repertoire: (this is Adobe Jenson Pro)

<code>\textsuperscript</code>	ab <sup>cdefghijklmnopq<sup>r</sup>stuv</sup> wxyz <sup>1234567890</sup>
<code>\textsubscript</code>	abc <sub>defghijklmnopq<sub>r</sub>stuv</sub> wxyz <sub>1234567890</sub>

The `[no-sscript]` package option will swap the definitions of the starred and non-starred versions of the commands described above if the new definitions are undesirable.

The macros `\realsubscript`, `\realsuperscript`, `\fakesubscript`, and `\fake-superscript` may be used to access the ‘new’ and ‘old’ functionalities regardless of the `[no-sscript]` package option.

## 2.2 Inner emphasis

`fixltx2e`’s method for checking for “inner” emphasis is a little fragile in  $\text{\TeX}$ , because font slant information might be missing from the font. Therefore, we use  $\text{\LaTeX}$ ’s `NESS` information, which is more likely to be correct.

---

<i>Nested <code>emphasis</code> is now <code>fixed</code>.</i>	<pre>\renewcommand\eminnershapes{\scshape} \fontspec{Didot} Nested {\em emphasis is         \emph{now} fixed.}</pre>
--	--

---

The `[no-emph]` package option will disable this redefinition.

## 2.3 Unicode footnote symbols

By default  $\text{\LaTeX}$  defines symbolic footnote characters in terms of commands that don’t resolve well; better results can be achieved by using specific unicode characters or proper LICRs with the `xunicode` package.

This problem has been solved by loading the `fixltx2e` and `xunicode` packages in `xltxtra`.

## 2.4 Verbatim

Many verbatim mechanisms assume the existence of a ‘visible space’ character that exists in the `ASCII` space slot of the typewriter font. This character is known in unicode as `U+2434: BOX OPEN`, which looks like this: ‘`□`’.

When a unicode typewriter font is used,  $\text{\LaTeX}$  no longer prints visible spaces for the `verbatim*` environment and `\verb*` command. `xltxtra` fixes this problem by using the correct unicode glyph, and patches the following packages to do the same: `listings`, `fancyvrb`, `moreverb`, and `verbatim`.

In the case that the typewriter font does not contain ‘`□`’, the Latin Modern Mono font is used as a fallback.

## 2.5 Logos

This part of the package essentially exists to define the `\XeTeX` and `\XeLaTeX` logos, which need to be tuned according to the font that is used. Originally I had some hard-coded definitions in here, but Andrew Moschou’s `metalogo` package now provides a much more flexible and useful interface to a variety of T<sub>E</sub>X-related logos.

Here are some examples. The default:

---

<code>\TeX</code> <code>\XeTeX</code> <code>\LaTeX</code> <code>\XeLaTeX</code>	<code>\TeX</code> <code>\XeTeX</code> <code>\LaTeX</code> <code>\XeLaTeX</code>
---	---

---

Notice that it’s a bit tight when not using Computer Modern, for which the logos were designed:

---

<code>\TeX</code> <code>\XeTeX</code> <code>\LaTeX</code> <code>\XeLaTeX</code>	<code>\usefont{OT1}{cmr}{m}{n}</code> <code>\TeX</code> <code>\XeTeX</code> <code>\LaTeX</code> <code>\XeLaTeX</code>
---	--

---

These logos, ideally, should be hand-tuned for each font that they’re used in. Please refer to the `metalogo` documentation for more information.

The `[no-logos]` package option will not redefine `\TeX` or `\LaTeX` but will still define `\XeTeX` and `\XeLaTeX`.

## 2.6 Discretionary hyphenation: `\-`

L<sup>A</sup>T<sub>E</sub>X defines the macro `\-` to insert discretionary hyphenation points. However, it is hard-coded in L<sup>A</sup>T<sub>E</sub>X to use the hyphen - character. Since `fontspec` makes it easy to change the hyphenation character on a per font basis, it would be nice if `\-` adjusted automatically — and now it does.

## 2.7 Vulgar fractions

The `\vfrac` command for setting ‘vulgar’ fractions based on AAT or OpenType font features. Not really recommended for many purposes, depending on your text, but it’s a good example of how to program such things using `fontspec`.

---

AAT: $\frac{123}{456}$ ICU: $\frac{123}{456}$	<code>\fontspec{Skia}</code> AAT: <code>\vfrac{123}{456}</code> <code>\fontspec{Warnock Pro}</code> ICU: <code>\vfrac{123}{456}</code>
--	---

---

(This can also be achieved in regular L<sup>A</sup>T<sub>E</sub>X with either the `nicefrac` or `xfrac` package.)

Only use it when you know it will work; no warnings are given if the font doesn't support the necessary features.

## 2.8 Named glyphs

Along the way somewhere, X<sub>Y</sub>TeX added support for selecting glyphs from a TrueType-based OpenType font based on their internal glyph name. Jonathan Kew posted the following definition as a nice interface to it.

---

¥ [smile]	<code>\fontspec{Charis SIL}</code>
	<code>\namedglyph{yen}</code>
	<code>\namedglyph{smile}</code>

---

## 2.9 The `\showhyphens` command

The default definition doesn't work in X<sub>Y</sub>TeX. A new version, written by Jonathan Kew, is included in this package that *does* work. Minor differences with the original: the showing of hyphens in the console output will be marked with explanatory text. Also, multiple words, separated by commas, will end up in separate instances of 'showing hyphens'.

## File I

# The xltextra package

This is the package implementation.

```
1 \ProvidesPackage{xltextra}
2 [2010/05/27 v0.5c Improvements for the "XeLaTeX" format]
```

Not for LuaTeX

```
3 \RequirePackage{ifluatex}
4 \ifluatex
5   \PackageWarningNoLine {xltextra} {^^]
6     XLTXTRA IS TO BE USED ONLY UNDER XETEX.
7     LOAD FONTSPEC DIRECTLY, INSTEAD.^^]
8     ABORTING LOADING%
9   }
10  \RequirePackage{fontspec}[2010/05/14 v2.0]
11  \expandafter \endinput
12 \fi
```

Required packages

```
13 \RequirePackage{ifxetex}
14 \RequireXeTeX
15 \RequirePackage{fontspec}[2010/05/14 v2.0]
16 \RequirePackage{xunicode}
```

Option processing

```
17 \newif\if@xxt@noscript@
18 \newif\if@xxt@nologos@
19 \newif\if@xxt@nohyphen@
20 \newif\if@xxt@noemph@
21 \newif\if@xxt@noverb@
22 \DeclareOption{no-sscript}{\@xxt@noscript@true}
23 \DeclareOption{no-logos}{\@xxt@nologos@true}
24 \DeclareOption{no-hyphen}{\@xxt@nohyphen@true}
25 \DeclareOption{no-emph}{\@xxt@noemph@true}
26 \DeclareOption{no-verb}{\@xxt@noverb@true}
27 \ProcessOptions*
```

### 3 Programming bits and pieces

#### 4 Logos

`\XeTeX` The T<sub>E</sub>X-related logos people insist upon using need to be tuned on a per-font basis.  
`\XeLaTeX` This package calls upon Andrew Moschou's package `metalogo` for this purpose. To tune the logos to each font, use the commands `\setlogokern`, `\setlogodrop`, etc. Refer to `mathspec`'s documentation for further details.

---

```
TeX XeTeX LaTeX XeLaTeX
LaTeX 2ε
\setlogokern{Xe}{-0.061em}
\setlogokern{eL}{-0.057em}
\setlogokern{La}{-0.265em}
\setlogokern{aT}{-0.0585em}
\setlogokern{Te}{-0.0575em}
\setlogokern{eX}{-0.072em}
\setlogokern{eT}{-0.056em}
\setlogokern{X2}{0.1667em}
\setlogodrop{0.153em}
\setLaTeXa{\scshape a}
\setLaTeXee{\mbox{\fontspec{Times}\itshape ε}}
\TeX\ \XeTeX\ \LaTeX\ \XeLaTeX\ \LaTeXe
```

---

```
28 \RequirePackage{metalogo}
```

The `[no-logos]` package option might be in effect, in which case `\TeX`, `\LaTeX` and `\LaTeXe` should keep their original definitions (which were saved by `metalogo`).

```
29 \if@xxt@nologos@
30 \let\TeX\original@TeX
31 \let\LaTeX\original@LaTeX
32 \let\LaTeXe\original@LaTeXe
33 \fi
```

`\TeX@logo@spacing` This macro is now deprecated. It is recommended to use the commands from `metalogo`.

```
34 \newcommand*\TeX@logo@spacing[6]{%
35 \PackageWarning{xltextra}{%
36 Use of \protect\TeX@logo@spacing\space is deprecated,\MessageBreak
37 recommend to use commands from package `metalogo' instead}
38 \setlogokern{Te}{#1}%
39 \setlogokern{eT}{#1}%
40 \setlogokern{eX}{#2}%
41 \setlogokern{Xe}{#2}%
42 \setlogodrop{#3}%
43 \setlogokern{La}{#4}%
```

```

44 \setlogokern{aT}{#5}%
45 \setlogokern{eL}{#6}}

```

## 4.1 Unicode footnote symbols

```

46 \RequirePackage{fixltx2e}[2006/03/24]

```

## 4.2 Emph

```

47 \unless\if@xxt@noemph@
\em  Redefinition of {\em ...} and \emph{...} to use NFSS info to detect when the
\emph inner shape should be used.
48 \DeclareRobustCommand\em
49   {\@nomath\em
50    \edef\@tempa{\f@shape}%
51    \edef\@tempb{\itdefault}%
52    \ifx\@tempa\@tempb
53      \emminnershape
54    \else
55      \emshape
56    \fi}
57 \DeclareTextFontCommand{\emph}{\em}
58 \let\emshape\itshape
59 \let\emminnershape\upshape
60 \fi

```

## 4.3 \-

```

61 \unless\if@xxt@nohyphen@
\ -  This macro is courtesy of Frank Mittelbach and the LATEX 2ε source code.
62 \DeclareRobustCommand{\-}{%
63   \discretionary{%
64     \char\ifnum\hyphenchar\font<\z@
65       \xlx@defaulthyphenchar
66     \else
67       \hyphenchar\font
68     \fi}{\}{\}}
69 \def\xlx@defaulthyphenchar{\-}
70 \fi

```

## 4.4 Subscript and superscript

For OpenType fonts, the subscript feature (subs) is used, but if that doesn't exist then the scientific inferior feature (sinf) is used on the assumption that something's better than nothing. This matches current trends in OpenType font design.



Footnotes are patched to use this better `\textsuperscript`.

```

\fake subscript The old ('fake') methods:
\fake superscript 71 \DeclareRobustCommand*\fake subscript[1]{%
72   \@text subscript{\selectfont#1}}
73 \DeclareRobustCommand*\fake superscript[1]{%
74   \@text superscript{\selectfont#1}}

\text subscript These commands are either defined to create fake or real sub-/super-scripts if they
\text subscript* are starred or not, respectively. This swaps if the [no-sscript] package option is
\text superscript in effect. Text subscripts:
\text superscript* 75 \if@xxt@nosscript@
76   \DeclareRobustCommand*\text subscript{%
77     \ifstar{\real subscript}{\fake subscript}}
78   \DeclareRobustCommand*\text superscript{%
79     \ifstar{\real superscript}{\fake superscript}}
80   \else
81     \DeclareRobustCommand*\text subscript{%
82       \ifstar{\fake subscript}{\real subscript}}
83     \DeclareRobustCommand*\text superscript{%
84       \ifstar{\fake superscript}{\real superscript}}
85   \fi

\real subscript
86 \ExplSyntaxOn
87 \DeclareRobustCommand*\real subscript[1]{
88   \fontspec_if_fontspec_font:TF
89   {
90     \fontspec_if_opentype:TF
91     {
92       \fontspec_if_feature:nTF {+subs}
93       {
94         {\addfontfeature{VerticalPosition=Inferior}#1}
95       }
96       {
97         \fontspec_if_feature:nTF {+sinf}
98         {
99           {\addfontfeature{VerticalPosition=ScientificInferior}#1}
100         }
101         {
102           \fake subscript{#1}
103         }
104       }
105     }

```

ATSUI fonts:

```
106 {
107   \fontspec_if_aat_feature:nnTF {10} {2}
108   {
109     {\addfontfeature{VerticalPosition=Inferior}#1}
110   }
111   {
112     \fakesubscript{#1}
113   }
114 }
115 }
```

Non-fontspec fonts:

```
116 {
117   \fakesubscript{#1}
118 }
119 }
```

\realsuperscript Text superscripts:

```
120 \DeclareRobustCommand*\realsuperscript[1]{
121   \fontspec_if_fontspec_font:TF
122   {
```

OpenType fonts:

```
123   \fontspec_if_opentype:TF
124   {
125     \fontspec_if_feature:nnTF {+sup}
126     {
127       {\addfontfeature{VerticalPosition=Superior}#1}
128     }
129     {
130       \fakesuperscript{#1}
131     }
132   }
```

ATSUI fonts:

```
133 {
134   \fontspec_if_aat_feature:nnTF {10} {1}
135   {
136     {\addfontfeature{VerticalPosition=Superior}#1}
137   }
138   {
139     \fakesuperscript{#1}
140   }
141 }
142 }
```

Non-fontspec fonts:

```

143 {
144   \fakesuperscript{#1}
145 }
146 }

```

Patching footnotes:

`\@makefnmark`

```

147 \def\@makefnmark{\mbox{\normalfont\textsuperscript{\@thefnmark}}}

```

`\vfrac` #1: Numerator

#2: Denominator

No error checking is done to ensure that the font actually has the necessary features. Requires the xunicode package for `\textfactionsolidus`.

```

148 \newcommand*\vfrac[2]{
149   \fontspec_if_fontspec_font:TF
150   {
151     \fontspec_if_opentype:TF
152     {
153       {\addfontfeature{VerticalPosition=Numerator}#1}
154       \textfactionsolidus
155       {\addfontfeature{VerticalPosition=Denominator}#2}
156     }
157     {
158       {\addfontfeature{VerticalPosition=Superior}#1}
159       \textfactionsolidus
160       {\addfontfeature{VerticalPosition=Inferior}#2}
161     }
162   }
163   {
164     \PackageError {xltextra}
165     { \string\vfrac\space~can~only~be~used~with~fontspec~fonts }
166     { Nothing~more~to~tell. }
167   }
168 }
169 \ExplSyntaxOff

```

`\namedglyph` #1: Name of the font glyph to be typeset

```

170 \newcommand\namedglyph[1]{%
171   \@tempcnta=\XeTeXglyphindex "#1"\relax
172   \ifnum\@tempcnta>0
173     \XeTeXglyph\@tempcnta
174   \else
175     \xxt@namedglyph@fallback{#1}%
176   \fi}

```

`\xxt@namedglyph@fallback` Redefine this macro to change how glyph names that aren't found get typeset.

```
177 \newcommand\xxt@namedglyph@fallback[1][[#1]]
```

`\showhyphens` This macro is entirely due to Jonathan Kew. I wish I knew how to write these sorts of things.

```
178 \newbox\xxt@tempbox
179 \def\showhyphens#1{%
180   \typeout{^^J*****}
181   \string\showhyphens:
182   *****}%
183   \@for\@ii:=#1\do{\xxt@showhyphens{\@ii}}%
184   \typeout{^^J*****}
185   *****%
186   *****^^J}}
187 \def\xxt@showhyphens#1{%
188   \setbox\@tempboxa=\vbox{%
189     \hsize1sp \hbadness10000 \hfuzz\maxdimen
190     \everypar={ } \leftskip\z@ \rightskip\leftskip
191     \pretolerance\m@ne \noindent \hskip\z@ #1\par
192     \global\setbox\xxt@tempbox=\hbox{\xxt@sh@cat}%
193     \setbox\@tempboxa=\hbox to \maxdimen{\unhbox\xxt@tempbox}}
194 \def\xxt@sh@cat{\unskip\unpenalty
195   \setbox\@tempboxa=\lastbox
196   \unless\ifvoid\@tempboxa
197     \global\setbox\xxt@tempbox=\hbox{%
198       \unhbox\@tempboxa
199       \unskip\unskip
200       \unhbox\xxt@tempbox}%
201   \expandafter\xxt@sh@cat
202   \fi}
```

## 4.5 Verbatims

Many thanks to Apostolos Syropoulos for discovering this problem and writing the redefinition of L<sup>A</sup>T<sub>E</sub>X's verbatim environment and `\verb*` command.

```
203 \unless\if@xxt@noverb@
```

`\xxt@visiblespace` Print U+2434: OPEN BOX, which is used to visibly display a space character.

```
204 \def\xxt@visiblespace{%
205   \iffontchar\font"2423
206   \expandafter\textvisiblespace
207   \else
208     \expandafter\xxt@visiblespace@fallback
209   \fi}
```

`\xxt@visiblespace@fallback` If the current font doesn't have u2434, use Latin Modern Mono instead.

```
210 \def\xxt@visiblespace@fallback{%
211   \usefont{EU1}{lmtt}{\f@series}{\f@shape}%
212   \textvisiblespace}}
```

`\xxt@vprintspaces` Helper macro to turn spaces active and print visible space instead.

```
213 \begingroup
214   \catcode\ =\active%
215   \gdef\xxt@vprintspaces{\catcode\ \active\let \xxt@visiblespace}%
216 \endgroup
```

`\verb` Redefine `\verb` to use `\xxt@vprintspaces`.

```
\verb* 217 \def\verb{\relax\ifmmode\hbox\else\leavevmode\null\fi
218   \bgroup
219   \verb@eol@error \let\do\@makeother \dospecials
220   \verbatim@font\@noligs
221   \@ifstar\@sverb\@verb}
222 \def\@sverb{\xxt@vprintspaces\@verb}
```

It's better to put small things into `\AtBeginDocument`, so here we go:

```
223 \AtBeginDocument{%
224   \xxt@patch@verbatim
225   \xxt@patch@moreverb
226   \xxt@patch@fancyvrb
227   \xxt@patch@listings}
```

`verbatim*` With the `verbatim` package.

```
228 \def\xxt@patch@verbatim{%
229   \@ifpackageloaded{verbatim}{%
230     \@namedef{verbatim*}{\begingroup\@verbatim\xxt@vprintspaces\verbatim@start}%
231   }{%
```

This is for vanilla LaTeX.

```
232   \@namedef{verbatim*}{\@verbatim\xxt@vprintspaces\@sxverbatim}%
233 }}
```

`listingcont*` This is for `moreverb`. The main `listing*` environment inherits this definition.

```
234 \def\xxt@patch@moreverb{%
235   \@ifpackageloaded{moreverb}{%
236     \@namedef{listingcont*}{%
237       \def\verbatim@processline{%
238         \thelisting@line \global\advance\listing@line1
239         \the\verbatim@line\par}%
240       \@verbatim\xxt@vprintspaces\verbatim@start}%
241     }{}}
```

listings and fancvrb make things nice and easy:

```
242 \def\xxt@patch@fancyvrb{%  
243   \@ifpackageloaded{fancyvrb}{%  
244     \let\FancyVerbSpace\xxt@visibleSPACE  
245   }{}}  
  
246 \def\xxt@patch@listings{%  
247   \@ifpackageloaded{listings}{%  
248     \let\lst@visibleSPACE\xxt@visibleSPACE  
249   }{}}
```

Finish verbatim features:

```
250 \fi
```