The XƎTEX reference guide

Will Robertson         Khaled Hosny
May 21, 2013

Introduction

This document serves to summarise additional features of XƎTEX without being so much as a ‘users’ guide. Note that much of the functionality addressed here is provided in abstracted form in various LƎTEX packages and ConTeXt modules.

The descriptions here should be a fairly exhaustive list of the new primitives and features of XƎTEX. Descriptions are still a little anemic, however.

License

This work, is distributed under the terms of the LaTeX Project Public License (http://www.latex-project.org/lppl.txt).

This basically means you are free to re-distribute this file as you wish; you may also make changes to this file or use its contents for another purpose, in which case you should make it clear, by way of a name-change or some other means, that your changed version is a modified version of the original. Please read the license text for more detailed information.
Contents

I \TeX specifics

1 The `\font` command
  1.1 Font options
  1.2 Font features
    1.2.1 Arbitrary OpenType, Graphite, or AAT features
    1.2.2 Options for all fonts
    1.2.3 OpenType script and language support
    1.2.4 Multiple Master and Variable Axes AAT font support
    1.2.5 Vertical typesetting

II New commands

2 Font primitives
  2.1 OpenType fonts
  2.2 AAT and Graphite fonts
    2.2.1 Features
    2.2.2 Feature selectors
    2.2.3 Variation axes
  2.3 Maths fonts

3 Character classes

4 Encodings

5 Line breaking

6 Graphics

7 Cross-compatibility with pdf\TeX\ and/or Lua\TeX

8 Misc.
Part I
\textsc{Xe\TeX\ specifics}

1 \textit{The} \texttt{\font} \textit{command}

Traditionally, fonts were selected in \TeX\ like this: \texttt{\font\1=⟨tfm name⟩} with various options possibly appended such as ‘at 10 pt’ or ‘scaled 1.2’, with obvious meaning. This syntax still works, but it has been greatly extended in Xe\TeX.

The extended syntax looks schematically like this:
\texttt{\font\1="⟨font identifier⟩⟨font options⟩:\langle font features⟩" (\TeX\ font options)}

The \texttt{⟨font identifier⟩} is the only mandatory part of the above syntax. If it is given in square brackets, (e.g., \texttt{[lmroman10-regular]}), it is taken as a font file name. Without brackets, the name is looked up both as a file name and a system font name. When using a font name, the font is looked up through the operating system, using (except on Mac OS X) the fontconfig library. Running \texttt{fc-list} should show you the font names available. E.g.,
\begin{verbatim}
\font\1="Liberation Serif" look for OS-installed font
\end{verbatim}

Fonts have many internal names, and Xe\TeX\ matches them in the following order:
\begin{itemize}
\item Full Name;
\item if the name has a hyphen, it is split into Family-Style pair then matched;
\item PostScript Name;
\item Family Name, if there is more than one match;
\begin{itemize}
\item look for font with ‘regular’ bit set in OS/2 table, if no match;
\item look for font with style “Regular”, “Plain”, “Normal” or “Roman”, in that order.
\end{itemize}
\end{itemize}

When using a file name, the xdvipdfmx driver must be used (this is the default). The current directory and the texmf trees are searched for files matching the name, or the path may be embedded in the font declaration, as usual with kpathsea. E.g.,
\begin{verbatim}
\font\2="[lmroman10-regular]" find lmroman10-regular.otf in any tree
\font\3="[/myfonts/fp9r8a]" look for fp9r8a only in /myfonts/
\end{verbatim}

A file with either an .otf, .ttf or .pfb extension (in that order) will be found. The extension can also be specified explicitly. If the file is a font collection (e.g., .ttc or .dfont), the index of the font can be specified using a colon followed by zero-based font index inside the square brackets. E.g.,
\begin{verbatim}
\font\4="[myfont.ttc:1]" load the second font from myfont.ttc file
\end{verbatim}
1.1 Font options

(Font options) are only applicable when the font is selected through the operating system (i.e., without square brackets). They may be any concatenation of the following:

- `/B` Use the bold version of the selected font.
- `/I` Use the italic version of the selected font.
- `/BI` Use the bold italic version of the selected font.
- `/IB` Same as `/BI`.
- `/S=x` Use the version of the selected font corresponding to the optical size \( x \) pt.
- `/AAT` Explicitly use the AAT renderer (Mac OS X only).
- `/OT` Explicitly use the OpenType renderer (new in 0.9999).
- `/GR` Explicitly use the Graphite renderer.\(^1\)
- `/ICU` Explicitly use the OpenType renderer (deprecated since 0.9999).

1.2 Font features

The (font features) is a comma or semi-colon separated list activating or deactivating various OpenType, Graphite, or AAT font features, which will vary by font. In contrast to font options, features work whether the font is selected by file name or through the operating system.

The \LaTeX{} documentation files `aat-info.tex` and `opentype-info.tex` provide per-font lists of supported features.

1.2.1 Arbitrary OpenType, Graphite, or AAT features

OpenType font features are chosen with standard tags\(^2\). They may be either comma- or semicolon-separated, and prefixed with a `+` to turn them on and a `-` to turn them off, optionally followed by `=` and a 0-based index for selecting alternates from multiple alternates features (ignored for `-` prefixed tags).

Example:

\begin{verbatim}
\font\liber="Linux Libertine O/I=5:+smcp" at 12pt
\liber This is the OpenType font Linux Libertine in italic with small caps.
\end{verbatim}

This is the OpenType font Linux Libertine in italic with small caps.

Varying depending on the language and script in use (see section §1.2.3 on page 6), a small number of OpenType features, if they exist, will be activated by default.

\(^1\)http://scripts.sil.org/cms/scripts/page.php?site_id=projects&item_id=graphite_home

\(^2\)http://www.microsoft.com/typography/otspec/featuretags.htm
\font\antt="Antykwa Torunska" at 12pt \antt 0
\font\antt="Antykwa Torunska:+aalt=0" at 12pt \antt 0
\font\antt="Antykwa Torunska:+aalt=1" at 12pt \antt 0
\font\antt="Antykwa Torunska:+aalt=2" at 12pt \antt 0
\font\antt="Antykwa Torunska:+aalt=3" at 12pt \antt 0
\font\antt="Antykwa Torunska:+aalt=4" at 12pt \antt 0

AAT font features and Graphite font features are specified by strings within each font rather than standardised tags. Therefore, even equivalent features between different fonts can have different names.

Example:
\font\gra="Charis SIL/GR:Small Caps=True" at 12pt
\gra This is the Graphite font Charis SIL with small caps.

**This is the Graphite font Charis SIL with small caps.**

1.2.2 Options for all fonts

Some font features may be applied for any font. These are

- **mapping=</font map>**
  Uses the specified font mapping for this font. This uses the TECKit engine to transform unicode characters in the last-minute processing stage of the source. For example, *mapping=tex-text* will enable the classical mappings from ugly ascii ``---'' to proper typographical glyphs “—”, and so on.

- **color=RRGGBB[TT]**
  Triple pair of hex values to specify the colour in RGB space, with an optional value for the transparency.

- **letterspace=\(x\)**
  Adds \(x/S\) space between letters in words, where \(S\) is the font size.

- **embolden=\(x\)**
  Increase the envelope of each glyph by the set amount (this makes the letters look ‘more bold’). \(x = 0\) corresponds to no change; \(x = 1.5\) is a good default value.

- **extend=\(x\)**
  Stretch each glyph horizontally by a factor of \(x\) (i.e., \(x = 1\) corresponds to no change).

- **slant=\(x\)**
  Slant each glyph by the set amount. \(x = 0\) corresponds to no change; \(x =
0.2 is a good default value. The slant is given by \( x = R/S \) where \( R \) is the displacement of the top edge of each glyph and \( S \) is the point size.

### 1.2.3 OpenType script and language support

OpenType font features (and font behaviour) can vary by script\(^3\) (‘alphabet’) and by language\(^4\). These are selected with four and three letter tags, respectively.

```latex
\texttt{script=\langle script \ tag\rangle}
```

Selects the font script.

```latex
\texttt{language=\langle lang \ tag\rangle}
```

Selects the font language.

### 1.2.4 Multiple Master and Variable Axes AAT font support

```latex
\texttt{weight=\langle x \rangle}
```

Selects the normalised font weight, \( x \).

```latex
\texttt{width=\langle x \rangle}
```

Selects the normalised font width, \( x \).

```latex
\texttt{optical size=\langle x \rangle}
```

Selects the optical size, \( x \) pt. Note the difference between the /S font option, which selects discrete fonts.

### 1.2.5 Vertical typesetting

```latex
\texttt{vertical}
```

Enables glyph rotation in the output so vertical typesetting can be performed.

### Part II

### New commands

### 2 Font primitives

```latex
\texttt{\%eTeXtracingfonts}
```

If nonzero, reports where fonts are found in the log file.

\(^3\) [http://www.microsoft.com/typography/otspec/scripttags.htm](http://www.microsoft.com/typography/otspec/scripttags.htm)

\(^4\) [http://www.microsoft.com/typography/otspec/languagetags.htm](http://www.microsoft.com/typography/otspec/languagetags.htm)
\texttt{\textbackslash XeTeXfonttype \langle font \rangle}
Expands to a number corresponding to which renderer is used for a \langle font \rangle:
0 for \LaTeX{} (a legacy TFM-based font);
1 for AAT;
2 for OpenType;
3 for Graphite.

<table>
<thead>
<tr>
<th>Example:</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{\newcommand\whattype[1]{% \texttt{\fontname#1} is rendered by \ifcase\XeTeXfonttype#1\TeX\or AAT\or OpenType\or Graphite\fi.\par} \font\1=&quot;cmr10&quot; \font\2=&quot;Charis SIL&quot; \font\3=&quot;Charis SIL/OT&quot; \whattype\1 \whattype\2 \whattype\3}</td>
</tr>
</tbody>
</table>
| \texttt{cmr10 is rendered by \LaTeX{}.
"Charis SIL" is rendered by Graphite.
"Charis SIL/OT" is rendered by OpenType.} |

\texttt{\textbackslash XeTeXfirstfontchar \langle font \rangle}
Expands to the code of the first character in \langle font \rangle.

\texttt{\textbackslash XeTeXlastfontchar \langle font \rangle}
Expands to the code of the last character in \langle font \rangle.

<table>
<thead>
<tr>
<th>Example:</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{\font\1=&quot;Charis SIL&quot;\1}</td>
</tr>
<tr>
<td>The first character in Charis SIL is: \texttt{&quot;\char\XeTeXfirstfontchar\1&quot;} and the last character is: \texttt{&quot;\char\XeTeXlastfontchar\1&quot;}.</td>
</tr>
<tr>
<td>\texttt{The first character in Charis SIL is: &quot; &quot; and the last character is: &quot;\textbackslash &quot;}</td>
</tr>
</tbody>
</table>

\texttt{\textbackslash XeTeXglyph \langle glyph slot \rangle}
Inserts the glyph in \langle glyph slot \rangle of the current font. Font specific, so will give different output for different fonts and possibly even different versions of the same font.

\texttt{\textbackslash XeTeXcountglyphs \langle font \rangle}
The count of the number of glyphs in the specified \langle font \rangle.
\texttt{\LaTeX}\TeX\emph{glyphname} \texttt{⟨font⟩} \texttt{⟨glyph slot⟩}

Expands to the name of the glyph in \texttt{⟨glyph slot⟩} of \texttt{⟨font⟩}. \textbf{Font specific}, so will give different output for different fonts and possibly even different versions of the same font.

\texttt{\LaTeX}\TeX\emph{glyphindex} "\texttt{⟨glyph name⟩}" \texttt{(space)} or \texttt{relax}

Expands to the glyph slot corresponding to the (possibly font specific) \texttt{⟨glyph name⟩} in the currently selected font. Only works for TrueType fonts (or TrueType-based OpenType fonts) at present. Use fontforge or similar to discover glyph names.

\texttt{\LaTeX}\TeX\emph{charglyph} \texttt{⟨char code⟩}

Expands to the default glyph number of character \texttt{⟨char code⟩} in the current font, or 0 if the character is not available in the font.

\textit{Example:}

\begin{verbatim}
\font\1="Charis SIL"

The glyph slot in Charis SIL for the Yen symbol is:
\texttt{\the\LaTeX\emph{glyphindex}"yen"} . % the font-specific glyph name
Or: \texttt{\the\LaTeX\emph{xcharglyph}"00A5"} . % the unicode character slot

This glyph may be typeset with the font-specific glyph slot:
\texttt{\LaTeX\emph{charglyph}150,}

\texttt{\LaTeX\emph{char"00A5}.}

\texttt{\LaTeX\emph{char"00A5}.}

The glyph slot in Charis SIL for the Yen symbol is: 150. Or: 150.

This glyph may be typeset with the font-specific glyph slot: ¥, or the unicode character slot:
¥.
\end{verbatim}

\texttt{\LaTeX}\TeX\emph{glyphbounds} \texttt{⟨edge⟩} \texttt{⟨glyph slot⟩}

Expands to a dimension corresponding to one of the bounds of a glyph, where \texttt{⟨edge⟩} is an integer from 1 to 4 indicating the left/top/right/bottom edge respectively, and \texttt{⟨glyph slot⟩} is an integer glyph index in the current font (only valid for non TFM-based fonts).

The left and right measurements are the glyph sidebearings, measured ‘inwards’ from the origin and advance respectively, so for a glyph that fits completely within its ‘cell’ they will both be positive; for a glyph that ‘overhangs’ to the left or right, they will be negative. The actual width of the glyph’s bounding box, therefore, is the character width (advance) minus both these sidebearings.

The top and bottom measurements are measured from the baseline, like \texttt{\TeX}’s height and depth; the height of the bounding box is the sum of these two dimensions.
Example:
\def\shadebbox#1{%
  \leavevmode\rlap{%
    \dimen0=\fontcharwd\font`#1%
    \edef\gid{\the\XeTeXcharglyph`#1}\
    \advance\dimen0 by -\XeTeXglyphbounds1 \gid
    \advance\dimen0 by -\XeTeXglyphbounds3 \gid
    \kern\XeTeXglyphbounds1 \gid
    \special{color push rgb 1 1 0.66667}\
    \vrule width \dimen0
    \kern\XeTeXglyphbounds3 \gid
    \special{color pop}\
    \kern\XeTeXglyphbounds3 \gid}
#1}
\noindent\font\x="Charis SIL/I" at 24pt \x
\shadebbox{A} \shadebbox{W} \shadebbox{a} \shadebbox{f}
\shadebbox{;} \shadebbox{*} \shadebbox{=}
\x

A W a f; * =

\XeTeXuseglyphmetrics
Counter to specify if the height and depth of characters are taken into account while typesetting (≥ 1). Otherwise (< 1), a single height and depth for the entire alphabet is used. Gives better output but is slower. Activated (≥ 1) by default.

Example:
\XeTeXuseglyphmetrics=0 \fbox{a}\fbox{A}\fbox{j}\fbox{J} vs.
\XeTeXuseglyphmetrics=1 \fbox{a}\fbox{A}\fbox{j}\fbox{J}

a A j J vs. a A j J

2.1 OpenType fonts

\XeTeXOTcountscripts \langle font \rangle
Expands to the number of scripts in the \langle font \rangle.
\XeTeXOTscripttag \(\text{font}\) \((\text{integer}, n)\)
Expands to the \(n\)-th script tag of \(\text{font}\).

\XeTeXOTcountlanguages \(\text{font}\) \((\text{script tag})\)
Expands to the number of languages in the script of \(\text{font}\).

\XeTeXOTlanguage\(\text{font}\) \((\text{script tag})\) \((\text{integer}, n)\)
Expands to the \(n\)-th language tag in the script of \(\text{font}\).

\XeTeXOTcountfeatures \(\text{font}\) \((\text{script tag})\) \((\text{language tag})\)
Expands to the number of features in the language of a script of \(\text{font}\).

\XeTeXOTfeature\(\text{font}\) \((\text{script tag})\) \((\text{language tag})\) \((\text{integer}, n)\)
Expands to the \(n\)-th feature tag in the language of a script of \(\text{font}\).

2.2 \textit{AAT and Graphite fonts}

2.2.1 \textit{Features}

\XeTeXcountfeatures \(\text{font}\)
Expands to the number of features in the \(\text{font}\).

\XeTeXfeaturecode \(\text{font}\) \((\text{integer}, n)\)
Expands to the feature code for the \(n\)-th feature in the \(\text{font}\).

\XeTeXfeaturename \(\text{font}\) \((\text{feature code})\)
Expands to the name corresponding to the \((\text{feature code})\) in the \(\text{font}\).

\XeTeXisexclusivefeature \(\text{font}\) \((\text{feature code})\)
Expands to a number greater than zero if the feature of a font is exclusive (can only take a single selector).

\XeTeXfindfeaturebyname \(\text{font}\) \((\text{feature name})\)
This command provides a method to query whether a feature name corresponds to a feature contained in the font. It represents an integer corresponding to the feature number used to access the feature numerically. If the feature does not exist, the integer is \(-1\). Also see \XeTeXfindselectorbyname.

\textit{Example:}
\begin{verbatim}
\font\1="Charis SIL/GR" at 10pt \\
def\featname{Uppercase Eng alternates} \\
The feature '{\featname}' has index \\
\the\XeTeXfindfeaturebyname\1 "\featname"\relax \\
The feature 'Uppercase Eng alternates' has index 1164863347
\end{verbatim}
2.2.2 Feature selectors

\texttt{\LaTeX\texttt{countselectors} \texttt{\langle font \rangle \langle feature \rangle}}

Expands to the number of selectors in a \texttt{\langle feature \rangle} of a \texttt{\langle font \rangle}.

\texttt{\LaTeX\texttt{selectorcode} \texttt{\langle font \rangle \langle feature code \rangle \langle integer, n \rangle}}

Expands to the selector code for the \texttt{n}-th selector in a \texttt{\langle feature \rangle} of a \texttt{\langle font \rangle}.

\texttt{\LaTeX\texttt{selectorname} \texttt{\langle font \rangle \langle feature code \rangle \langle selector code \rangle}}

Expands to the name corresponding to the \texttt{\langle selector code \rangle} of a feature of a \texttt{\langle font \rangle}.

\texttt{\LaTeX\texttt{isdefaultselector} \texttt{\langle font \rangle \langle feature code \rangle \langle selector code \rangle}}

Expands to a number greater than zero if the selector of a feature of a font is on by default.

\texttt{\LaTeX\texttt{findselectorbyname} \texttt{\langle font \rangle \langle feature name \rangle \langle selector name \rangle}}

This command provides a method to query whether a feature selector name corresponds to a selector of a specific feature contained in the font. It represents an integer corresponding to the selector number used to access the feature selector numerically. If the feature selector does not exist, the integer is -1.

The indices given by this command and by \texttt{\LaTeX\texttt{findfeaturebyname}} can be used in Graphite fonts to select font features directly (see example below). Alternatively, they can be used as a means of checking whether a feature-selector exists before attempting to use it.

\textbf{Example:}

\texttt{\font\texttt{1}="Charis SIL/GR" at 10pt}
\texttt{\def\featname{"Uppercase Eng alternates"}}
\texttt{\newcount\featcount}
\texttt{\featcount=\LaTeX\texttt{findfeaturebyname}\texttt{1 "\featname"\relax}}

\texttt{\def\selecname{"Large eng on baseline"}}
\texttt{\newcount\seleccount}
\texttt{\seleccount=\LaTeX\texttt{findselectorbyname}\texttt{1 \featcount "\selecname"\relax}}

The feature selector \texttt{"selecname'} has index \texttt{\the\seleccount}

\texttt{\font\texttt{2}="Charis SIL/GR:\featname=\selecname" at 10pt}
\texttt{\font\texttt{3}="Charis SIL/GR:\the\featcount=\the\seleccount" at 10pt}

Activating the feature: \texttt{\1 Ŋ \2 Ŋ \3 Ŋ}

The feature selector \texttt{"Large eng on baseline'} has index 1

Activating the feature: \texttt{ŋ Ŋ Ŋ}
2.2.3 Variation axes

\XeTeXcountvariations ⟨font⟩
Expands to the number of variation axes in the ⟨font⟩.

\XeTeXvariation ⟨font⟩ ⟨integer, n⟩
Expands to the variation code for the n-th feature in the ⟨font⟩.

\XeTeXvariationname ⟨font⟩ ⟨variation code⟩
Expands to the name corresponding to the ⟨feature code⟩ in the ⟨font⟩.

\XeTeXvariationmin ⟨font⟩ ⟨variation code⟩
Expands to the minimum value of the variation corresponding to the ⟨variation code⟩ in the ⟨font⟩.

\XeTeXvariationmax ⟨font⟩ ⟨variation code⟩
Expands to the maximum value of the variation corresponding to the ⟨variation code⟩ in the ⟨font⟩.

\XeTeXvariationdefault ⟨font⟩ ⟨variation code⟩
Expands to the default value of the variation corresponding to the ⟨variation code⟩ in the ⟨font⟩.

\XeTeXfindvariationbyname ⟨font⟩ ⟨variation name⟩
An integer corresponding to the internal index corresponding to the ⟨variation name⟩. This index cannot be used directly but may be used to error-check that a specified variation name exists before attempting to use it.

2.3 Maths fonts

The primitives described following are extensions of \TeX’s 8-bit primitives.

In the following commands, ⟨fam.⟩ is a number (0–255) representing font to use in maths. ⟨math type⟩ is the 0–7 number corresponding to the type of math symbol; see a \TeX reference for details.

Before version 0.9999.0 the following primitives had \XeTeX prefix instead of \U, the old names are deprecated and will be removed in the future.

\Umathcode ⟨char slot⟩ [=] ⟨math type⟩ ⟨fam.⟩ ⟨glyph slot⟩
Defines a maths glyph accessible via an input character. Note that the input takes three arguments unlike \TeX’s \mathcode.
\texttt{\textbackslash Umathcodenum} \langle char \ slot \rangle \ [\ifnum\mathcode=] \langle \text{math type/fam./glyph \ slot} \rangle

Pure extension of \texttt{\textbackslash mathcode} that uses a ‘bit-packed’ single number argument. Can also be used to extract the bit-packed mathcode number of the \langle char slot \rangle if no assignment is given.

\texttt{\textbackslash Umathchar} \langle \text{math type} \rangle \ (\text{fam.}) \ (\text{glyph slot})

Typesets the math character in the \langle glyph slot \rangle in the family specified.

\texttt{\textbackslash Umathcharnum} \langle \text{type/fam./glyph \ slot} \rangle

Pure extension of \texttt{\textbackslash mathchar} that uses a ‘bit-packed’ single number argument. Can also be used to extract the bit-packed mathcode number of the \langle char slot \rangle if no assignment is given.

\texttt{\textbackslash Umathchardef} \langle \text{control sequence} \rangle \ [\ifnum\mathchar=] \langle \text{math type} \rangle \ (\text{fam.}) \ (\text{glyph slot})

Defines a maths glyph accessible via a control sequence.

\texttt{\textbackslash Umathcharnumdef} \langle \text{control sequence} \rangle \ [\ifnum\mathchar=] \langle \text{type/fam./glyph \ slot} \rangle

Defines a control sequence for accessing a maths glyph using the ‘bit-packed’ number output by, e.g., \texttt{\textbackslash Umathcodenum}. This would be used to replace legacy code such as \texttt{\textbackslash mathchardef \foo=\textbackslash mathcode \backslash}. 

\texttt{\textbackslash Udelcode} \langle char \ slot \rangle \ [\ifnum\delcode=] \langle \text{fam.} \rangle \ (\text{glyph slot})

Defines a delimiter glyph accessible via an input character.

\texttt{\textbackslash Udelcodenum} \langle char \ slot \rangle \ [\ifnum\delcode=] \langle \text{fam.} \rangle \ (\text{glyph slot})

Pure extension of \texttt{\textbackslash delcode} that uses a ‘bit-packed’ single number argument. Can also be used to extract the bit-packed delcode number of the \langle char slot \rangle if no assignment is given.

\texttt{\textbackslash Udelimiter} \langle \text{math type} \rangle \ (\text{fam.}) \ (\text{glyph slot})

Typesets the delimiter in the \langle glyph slot \rangle in the family specified of either \langle \text{math type} \rangle 4 (opening) or 5 (closing).

\texttt{\textbackslash Umathaccent [ \text{keyword} ]} \langle \text{math type} \rangle \ (\text{fam.}) \ (\text{glyph slot})

Typesets the math accent character in the \langle glyph slot \rangle in the family specified. Starting from version 0.9998, \texttt{\textbackslash Umathaccent} accepts optional keyword:

**fixed**  Don’t stretch the accent, the default is to stretch it: $\hat{M}$ vs $\hat{\hat{M}}$.

**bottom**  Place the accent below its base. Can be followed by the **fixed** keyword.

\texttt{\textbackslash Uradical} \langle \text{fam.} \rangle \ (\text{glyph slot})

Typesets the radical in the \langle glyph slot \rangle in the family specified.
3 Character classes

The idea behind character classes is to define a boundary where tokens can be added to the input stream without explicit markup. It was originally intended to add glue around punctuation to effect correct Japanese typesetting. This feature can also be used to adjust space around punctuation for European traditions. The general nature of this feature, however, lends it to several other useful applications including automatic font switching when small amounts of another language (in another script) is present in the text.

\texttt{\textbackslash \texttt{XeTeXinterchartokenstate}}
Counter. If positive, enables the character classes functionality.

\texttt{\textbackslash \texttt{newXeTeXintercharclass}} \langle control sequence \rangle
Allocates a new interchar class and assigns it to the \langle control sequence \rangle argument.

\texttt{\textbackslash \texttt{XeTeXcharclass}} \langle char slot \rangle [\texttt{=} \langle interchar class \rangle]
Assigns a class corresponding to \langle interchar class \rangle (range 0–255) to a \langle char slot \rangle. Most characters are class 0 by default. Class 1 is for CJK ideographs, classes 2 and 3 are CJK punctuation. The boundary of a text string is considered class 255, wherever there is a boundary between a ‘run’ of characters and something else — glue, kern, math, box, etc. Special case class 256 is ignored; useful for diacritics so I’m told.

\texttt{\textbackslash \texttt{XeTeXinterchartoks}} \langle interchar class 1 \rangle \langle interchar class 2 \rangle [\texttt{=} \{ \langle token list \rangle \}]
Defines tokens to be inserted between \langle interchar class 1 \rangle and \langle interchar class 2 \rangle (in that order).
Example:
\XeTeXinterchartokenstate = 1
\newXeTeXintercharclass \mycharclassa
\newXeTeXintercharclass \mycharclassA
\newXeTeXintercharclass \mycharclassB
\XeTeXcharclass \a \mycharclassa
\XeTeXcharclass \A \mycharclassA
\XeTeXcharclass \B \mycharclassB

% between "a" and "A":
\XeTeXinterchartoks \mycharclassa \mycharclassA = {\itshape}
\XeTeXinterchartoks \mycharclassA \mycharclassa = {\upshape}

% between " " and "B":
\XeTeXinterchartoks 255 \mycharclassB = {\bgroup\color{blue}}
\XeTeXinterchartoks \mycharclassB 255 = {\egroup}

% between "B" and "B":
\XeTeXinterchartoks \mycharclassB \mycharclassB = {.}

aAa A a B aBa BB

a[A]a A a B aBa B.B

In the above example the input text is typeset as
a[\itshape A\upshape]a A a \bgroup\color{blue}B\egroup aBa B.B
4 Encodings

\XeTeXinputnormalization (Integer)
Specify whether XƎTEX is to perform normalisation on the input text and, if so, what type of normalisation to use. See http://unicode.org/reports/tr15/ for a description of Unicode normalisation. The (Integer) value can be:

0 (default) do not perform normalisation.
1 normalise to NFC form, using precomposed characters where possible instead base characters with combining marks.
2 normalise to NFD form, using base characters with combining marks instead of precomposed characters.

\XeTeXinputencoding (Charset name)
Defines the input encoding of the following text.

\XeTeXdefaultencoding (Charset name)
Defines the input encoding of subsequent files to be read.

5 Line breaking

\XeTeXdashbreakstate (Integer)
Specify whether line breaks after en- and em-dashes are allowed. Off, 0, by default.

\XeTeXlinebreaklocale (Locale ID)
Defines how to break lines for multilingual text.

\XeTeXlinebreakskip (Glue)
Inter-character linebreak stretch

\XeTeXlinebreakpenalty (Integer)
Inter-character linebreak penalty

\XeTeXupwardsmode (Integer)
If greater than zero, successive lines of text (and rules, boxes, etc.) will be stacked upwards instead of downwards.
6 Graphics

Thanks to Heiko Oberdiek, Paul Isambert, and William Adams for their help with the documentation in this section.

\texttt{\LaTeX{}picfile} \langle \texttt{filename} \rangle \begin{array}{llllll}
\left[ \text{ scaled } \langle \texttt{int} \rangle \mid \text{ xscaled } \langle \texttt{int} \rangle \mid \text{ yscaled } \langle \texttt{int} \rangle \mid
\right. \\
\left. \text{ width } \langle \texttt{dimen} \rangle \mid \text{ height } \langle \texttt{dimen} \rangle \mid \text{ rotated } \langle \texttt{decimal} \rangle \right] 
\end{array}

Insert an image. See below for explanation of optional arguments.

\texttt{\LaTeX{}pdffile} \langle \texttt{filename} \rangle \begin{array}{llllllll}
\left[ \text{ page } \langle \texttt{int} \rangle \right. \\
\left. \mid \text{ crop } \mid \text{ media } \mid \text{ bleed } \mid \text{ trim } \mid \text{ art } \right]
\end{array}
\begin{array}{llllllll}
\left[ \text{ scaled } \langle \texttt{int} \rangle \mid \text{ xscaled } \langle \texttt{int} \rangle \mid \text{ yscaled } \langle \texttt{int} \rangle \mid \text{ width } \langle \texttt{dimen} \rangle \mid
\right. \\
\left. \text{ height } \langle \texttt{dimen} \rangle \mid \text{ rotated } \langle \texttt{decimal} \rangle \right] 
\end{array}

Insert (pages of) a PDF. See below for explanation of optional arguments.

In the graphic/PDF commands above, \langle \texttt{filename} \rangle is the usual file name argument of \texttt{\LaTeX{}}\input, \texttt{\LaTeX{}}\openin, etc. It must not terminated by \texttt{\LaTeX{}}\relax if options are given. \langle \texttt{int} \rangle and \langle \texttt{dimen} \rangle are the usual integer or dimen specifications of regular \LaTeX{}.

The rotation is specified in degrees (i.e., an input of ‘360’ is full circle) and the rotation is counterclockwise. The syntax of \langle \texttt{decimal} \rangle require some explanation:

\langle \texttt{decimal} \rangle \rightarrow \langle \texttt{optional signs} \rangle \langle \texttt{unsigned decimal} \rangle
\langle \texttt{unsigned decimal} \rangle \rightarrow \langle \texttt{normal decimal} \rangle \mid \langle \texttt{coerced dimen} \rangle \mid \langle \texttt{internal dimen} \rangle
\langle \texttt{normal decimal} \rangle \rightarrow \langle \texttt{normal integer} \rangle \mid \langle \texttt{decimal constant} \rangle

A \langle \texttt{coerced dimen} \rangle or \langle \texttt{internal dimen} \rangle is interpreted as number with unit ‘pt’. For example, for a rotation specified with a dimension \texttt{\LaTeX{}}\testdim,

\begin{itemize}
  \item \texttt{\LaTeX{}}\testdim=45pt results in a rotation of 45°,
  \item \texttt{\LaTeX{}}\testdim=1in is 72.27°, and
  \item \texttt{\LaTeX{}}\testdim=100sp is (100/65536)°.
\end{itemize}

In all cases the resulting decimal number for rotation \(x\) must be within the limits \(-16384 < x < 16384\).

The \texttt{\LaTeX{}pdffile} command takes one more optional argument for specifying to which ‘box’ the PDF should be cropped before inserting it (the second optional argument listed in the syntax of \texttt{\LaTeX{}pdffile} above). The PDF standard defines a number of (rectangular) bounding boxes that may be specified for various purposes. These are described in the PDF Standard\footnote{Adobe Systems Incorporated, 2008: \url{http://www.adobe.com/devnet/acrobat/pdfs/PDF32000_2008.pdf}} and summarised below.

\texttt{media} the box defining the physical page size.

crop  the box of the page contents for display/printing purposes.
bleed the box containing the page contents plus whatever extra space required for printing purposes.
trim  the box of the finished page after trimming the printed ‘bleed box’.
art   the box containing the ‘meaningful content’ of the page. This could be the crop box with boilerplate text/logos trimmed off.

When not specified in the PDF to be inserted, the crop box defaults to the media box, and the bleed, trim, and art boxes default to the crop box.

\XeTeXpdfpagecount (filename)
Expands to the number of pages in a PDF file.

7 Cross-compatibility with pdfTEX and/or LuaTeX

\pdfpageheight ⟨dimension⟩
The height of the PDF page.

\pdfpagewidth ⟨dimension⟩
The width of the PDF page.

\pdfsavepos
Saves the current location of the page in the typesetting stream.

\pdflastxpos
Retrieves the horizontal position saved by \pdfsavepos.

\pdflastypos
Retrieves the vertical position saved by \pdfsavepos.

\ifincsname...(\else...\fi
TeX conditional to branch true if the expansion occurs within \csname ... \endcsname.

Example:
\def\x{\ifincsname y\else hello\fi}
\def\y{goodbye}
\x/\csname x\endcsname

hello/goodbye

18
\ifprimitive\langle control sequence\rangle...\langle\else...\rangle\fi
\TeX\ conditional to test if a control sequence is a primitive and that it has not been
redefined.

\primitive\langle control sequence\rangle
If the control sequence is a primitive that’s been redefined, this command causes it to
expand with its original (i.e., primitive) definition.

\shellescape
Read-only status indicating the level of ‘shell escape’ allowed. That is, whether com-
mands are allowed to be executed through \write18\{...\}. Expands to zero for off;
one for on (allowed); two is ‘restricted’ (default in TeX Live 2009 and greater) in which
a subset of commands only are allowed.

Example:
Shell escape \ifnum\shellescape>0 is \else is not \fi enabled.
Shell escape is enabled.

\strcmp\langle arg one\rangle\langle arg two\rangle
Compares the full expansion of the two token list arguments. Expands to zero if they
are the same, less than one if the first argument sorts lower (lexicographically) than
the second argument, and greater than one if vice versa.

Example:
‘a’ is less than ‘z’: \strcmp{a}{z}
\def\z{a}
The tokens expand before being compared: \strcmp{a}{\z}
\def\a{z}
Therefore, |\a| is greater than |\z|: \strcmp{\a}{\z}
\edef\b{\string b}
Also note that catcodes are ignored: \strcmp{b}{\b}
‘a’ is less than ‘z’: -1
The tokens expand before being compared: 0
Therefore, \a is greater than \z: 1
Also note that catcodes are ignored: 0
\suppressfontnotfounderror (integer)
When set to zero (default) if a font is loaded that cannot be located by \TeX, an error message results and typesetting is halted. When set to one, this error message is suppressed and the font control sequence being defined is set to \nullfont.

Example:
\suppressfontnotfounderror=1
\font\x="ImpossibleFont" at 10pt
\ifx\x\nullfont
  \font\x="Georgia" at 10pt
\fi
\x This would be ‘ImpossibleFont’, if it existed.

8 Misc.

\XeTeXversion
Expands to a number corresponding to the \TeX version.

\XeTeXrevision
Expands to a string corresponding to the \TeX revision number.

Example:
The \XeTeX version used to typeset this document is:
\the\XeTeXversion\XeTeXrevision

The \TeX version used to typeset this document is: 0.9999