

X_YT_EX 中文排版: *zhspacing* 用户参考手册^{*}

YIN Dian[†][著] YANG Cheng[‡][译]

2010/7/20

目录

I	zhspacing.sty	1
1	简介	1
2	开始	1
3	高级用法	3
3.1	字体	3
3.2	更多关于字体	4
3.3	空白	5
3.4	中文竖排	6
3.5	兼容性	7
3.6	与 ctex 包同时使用	7

^{*}参考官方文档翻译 (版本 2007/12/11), 如有不妥, 请以官方文献为准。

[†]Lab. of Network Comm. System & Control, Univ. of Sci. & Tech. of China. 中国科学技术大学。

[‡]Mathematics & Applied Math, Univ. of Sci. & Tech. of Beijing. 北京科技大学。E-mail: ustbyangcheng@gmail.com。

4 获得 zhspacing	8
II zhfont.sty	8
1 简介	8
2 包选项	8
3 宏支持	9
4 例子	9
III zhmath.sty	9
1 简介	10
2 包选项	10
3 宏支持	10
4 例子	11

文件 I

zhspacing.sty

1 简介

X_ƎT_EX 是由 Jonathan Kew 和 SIL International 编写的一个新的 T_EX 引擎，该引擎可以支持普通的 Unicode 和高级字体的 ϵ -T_EX¹。使用 X_ƎT_EX 可以很方便的排版任何语言的所有使用中的字符，并且不需要任何其他的包，如 CJK²。然而，X_ƎT_EX 自身并不能很好的解决所有的这些问题。其中中文排版的一些细节问题在 X_ƎT_EX 中并没有被处理，如在中文和西文字符间自动选择字体，忽略全角符号调整，自动忽略在中西文或数学公式中插入的空格，等等。为了使 X_ƎT_EX 能够实现简单和高质量的中文文档的排版，宏包 *zhspacing* 被编写出来，它利用了 X_ƎT_EX 0.997 版本提供的原语 `\XeTeXinterchartoks`。

2 开始

zhspacing 能够在 plain X_ƎT_EX 或 X_ƎL_AT_EX 中使用，使用方法也相当简单。例如，在 plain X_ƎT_EX 中，

```
\input zhspacing.sty
\zhspacing
这是中文测试。中文和English的混排。中文和$E=mc^2$的混排。
\bye
```

在 X_ƎL_AT_EX 中，

```
\documentclass{article}
\usepackage{zhspacing}
\zhspacing
```

¹这是从 ConTeXt Wiki 拷贝的。

²CJK 是一个 L_AT_EX 2_ε 的一个宏包，它使得 CJK 脚本(Chinese/Japanese/Korean)具有多种编码，由 Werner Lemberg 编写。

```
\begin{document}
这是中文测试。中文和English的混排。中
文和 $E=mc^2$ 的混排。
\end{document}
```

上面这两个例子产生如下相同的结果:

这是中文测试。中文和 English 的混排。中文和 $E = mc^2$ 的混排。

正如你所看到的，中文后的空格总是被忽略。此外，一个值得关注的空白在中西文中同样在数学公式中被插入。实际上，下面所有的输入可以产生混合语言的输出，在中西文之间的空白会自动被插入，

中Eng文， 中Eng文， 中Eng 文 and 中Eng 文.

它们相应的输出是，

中 Eng 文， 中 Eng 文， 中 Eng 文 and 中 Eng 文.

仔细观察，你将发现，第一个输入和第二个恰好有相同的输出，第三、四个也是一样。然而，最后两个例子在 Eng 和文之间的空白比中 $和Eng$ 要宽一些。这是因为空格是由字母g后的空格记号生成的，并不是由zhspacing的机制自动插入的。zhspacing自动插入使得空格被忽略和跳过的功能使得写文档更加舒适。你不需要担心手工插入空白字符，例如在CJK中加入烦人的 \sim 。现在回到标点符号空白调整的主题。正式的中文排版需要连续的全角标点被压缩，并且在全角符号前后的删除线将切断该符号的空格，使得它对齐到页边的空白处。zhspacing能够解决上述问题，也就是有适当的禁则。这里有个例子。

他强调，“三个代表”重要思想是在新的历史条件下运用马克思主义的立场、观点和方法的典范，是我们学习马克思主义的立场、观点和方法最现实、最生动的教材。“三个代表”重要思想是与时俱进的理论。

3 高级用法

3.1 字体

zhspacing 使用一种可拓展的方式来选择字体。规则总结如下，

- 西文字体，如对于既不是表意文字和标点的 CJKV，默认使用的是相同的字体。
- 中文字体使用独立的字体。文档中字体的改变并不会影响到已使用的字体去显示中文，除非你使用 NFSS 并且改变了字体系列或形状。
- 在显示基本的中文表意文字时，执行命令`\zhfont`。
- 在显示中文标点时，执行命令`\zhpunctfont`。
- 在显示 CJK Ext-A 字符时，执行命令`\zhcjkextafont`。
- 在显示 CJK Ext-B 字符时，执行命令`\zhcjkextbfont`。
- 当从非中文字符转换到中文字符时，执行`\zhs@savefont`。当转换回来，执行`\zhs@restorefont`。

当从 plain \TeX 中导入 zhspacing，上述命令是默认定义的，

```
\font\zhfont="SimSun" at 10pt
\font\zhpunctfont="SimSun" at 10pt
\def\zhcjkextafont{\message{CJK Ext-A}}
\def\zhcjkextbfont{\message{CJK Ext-B}}
\let\zhs@savefont=\begingroup
\let\zhs@restorefont=\endgroup
```

当从 \LaTeX 中导入时，默认定义如下，

```
\newfontfamily\zhfont[BoldFont=SimHei]{SimSun}
\newfontfamily\zhpunctfont{SimSun}
\def\zhcjkextafont{\message{CJK Ext-A}}
\def\zhcjkextbfont{\message{CJK Ext-B}}
\def\zhs@savefont{\zhs@savefont{old}}
\def\zhs@restorefont{\zhs@restorefont{old}}
```

这里`\zhs@savefont` 和 `\zhs@restorefont` 是内置的宏，用于保存和恢复当前字体的 NFSS 信息到指定的位置。考虑到不是每个用户都安装特殊字体，CJK Ext-A/B 字体默认情况下并没有定义。我建议使用`Sun-ExtA`和`Sun-ExtB`作为相应的字体。你可以用定义`\zhfont` 的相似的方式自己去定义额外的字体宏。

一个多种不同类的字符的混合排版的例子如下，引自 *The Tale of Kiều*³。

些 Trăm năm, trong cõi người ta, 才
命 罗怙 Chữ tài, chữ mệnh, khéo là ghét nhau.
戈 局 櫟 Trãi qua một cuộc bể dâu, 仍條
疸 Những điều trông thấy mà đau đớn lòng. ⁴

3.2 更多关于字体

有时，在 L^AT_EX 中对于不同的 NFSS 字体家族需要用不同的中文字体来描述。这个可以通过改变`\zhs@savefont` 和 `\zhfont` 的定义来实现，并不需要修改 `zhspacing` 源文件。这是一种方式⁵

```
\makeatletter
\g@addto@macro\zhs@savefont{%
  \long\edef\zhs@tmpmacro{\f@family}%
  \def\zhs@curr@fam{0}%
  \ifx\zhs@tmpmacro\sfdefault
    \def\zhs@curr@fam{1}%
  \else\ifx\zhs@tmpmacro\ttdefault
    \def\zhs@curr@fam{2}%
  \fi\fi
  \edef\zhs@tmpmacro{\f@family}%
  \ifx\zhs@tmpmacro\sfdefault
    \def\zhs@curr@fam{1}%
  \else\ifx\zhs@tmpmacro\ttdefault
    \def\zhs@curr@fam{2}%
  \fi\fi
}
```

³《金云翘传》，或 *Truyện Kiều*，大部分用 CJK Ext-A/B 编码，甚至不包含 Unicode 编码。

⁴由于字符集的缺失，一些表意文字被相似的字符替换。原手册的图片可见 [this link](#).

⁵一种更加方便的方式请参看后面的 `zhfont.sty`

```

\newfontfamily\zhrmfont[BoldFont=SimHei,
                        ItalicFont=KaiTi_GB2312]{SimSun}
\newfontfamily\zhsffont{SimHei}
\newfontfamily\zh.ttf[BoldFont=SimHei]
                        {KaiTi_GB2312}
\def\zhfont{\ifcase\zhs@curr@fam\zhrmfont]or
            \zhsffont]or
            \zh.ttf\else\zhrmfont\fi}

```

3.3 空白

zhspacing 中的空白机制是灵活的。zhspacing 利用下面的空白命令。注意它们被定义成宏而不是空白指令的值，为了使空白可以根据不同的字体大小改变。

`\skipzh` 在紧邻的中文字符间空白

`\skipenzh` 在中西文字符间或公式间空白

`\skipzhopen` 在开始的全角符号前的空白，如““”，“（”，“《”，等

`\skipzhinteropen` 当优于另一个全角符号的全角符号前的空白

`\skipzhlinestartopen` 当全角符号在一行的开头时的空白

`\skipzhclose` 在闭合的全角符号后的空白，如“””，“）”，“》”等

`\skipzhinterclose` 当跟在另一个全角符号后的全角符号后的空白

`\skipzhlineendclose` 当全角符号在行尾使得空白

`\skipzhfullstop` 在全停的符号后的空白，如“；”，“．”，“。”，等

`\skipzhinterfullstop` 当跟在另一个全角符号后的全停符号后的空白

`\skipzhlineendfullstop` 当全停符号出现在行尾时的空白

`\skipzhhalfstop` 在半角符号后的空白，如“、”，“，”，“：”，等

`\skipzhinterhalfstop` 当跟在另一个等宽的符号后的等宽半角符号后的空白

`\skipzhlineendhalfstop` 出现在行尾的等宽的半角符号后的空白

`\skipnegzhlinestartopen` 为 `\skipzhlinestartopen` 相反的空白

`\skipnegzhlineendclose` 为 `\skipzhlineendclose` 相反的空白

`\skipnegzhlineendfullstop` 为 `\skipzhlineendfullstop` 相反的空白

`\skipnegzhlineendhalfstop` 为 `\skipzhlineendhalfstop` 相反的空白

上述所有的空白命令都是用下面的模式定义的

```
\def\skipxxx{\hskip xxxxx}.
```

`zhspacing` 有三个预定义的空白方案,即 `\simsunskipscheme`, `\emptyskipscheme` 和 `\haltskipscheme`。第一个方案适用于 SimSun 字体和其他流行的中文字体,但它不支持 OpenType 的 `halt` 特性,和在开的符号前及在闭合或句逗号后需要负的空白。第二种简单的加上了零长度。最后一种这对 OpenType 中文字体的 `halt` 特性提供了支持,如 Adobe Song Std, 这里就应该在确定的符号前后插入正的空白。当然你也可以自定义自己的空白方案。

3.4 中文竖排

对于特殊的字体,可以通过增加粗糙的特性 `vertical` 来完成中文的竖排。例子如下,

```
\documentclass[12pt]{article}
\usepackage[dvipdfm]{graphicx}
\usepackage{zhspacing}
\zhspacing
\begin{document}
\newfontfamily\zhfont[RawFeature={vertical:}]
                        {SimSun}
\newfontfamily\zhpunctfont[RawFeature=
                        {vertical:+vert:+vhal}]
                        {[AdobeSongStd-Light.otf]}
\haltskipscheme
\rotatebox{-90}{我是中国人，我爱自己的祖国。}
\end{document}
```

我是中国人，我爱自己的祖国。

对于上面的例子，我们应该注意，为了有合适的竖排的符号，我们设置`\zhpunctfont` 使用 Adobe 支持的 `vert` 特性，并且改变空白方案为`\haltskipsscheme` 以匹配特定的 `vhal` 特性。然而，在排版中使用某些的字体，如`KaiTi_GB2312`等，且包含符号的竖排中文仍然存在一些 bug。5 月份我已经将 bug 报告给了jjgod，但似乎他还没有解决。更进一步，竖排中文的基线也不正确，因此竖排模式下中英文混合排版会出现一些不好的结果，我们需要避免。

3.5 兼容性

理论上，`zhspacing` 应该和所有的宏包都兼容，除非在某些特殊的应用中谁更改了`\hskip` 和`\penalty` 的定义。当使用公共的宏包，如 `hyperref` 和 `fancyhdr`，我还没有发现任何的冲突。然而，`ulem` 是`\hskip` 和`\penalty` 的重新定义，并且会导致不期望的输出⁶。请使用 `zhspacing` 提供的 `zhulem` 替代。

3.6 与 ctex 包同时使用

`ctex` 包，由`ctex.org`制作，广泛用于中文文档框架的处理，它对用户隐藏了中文处理的细节，使得在不同的中文处理包或系统之间处理的方便，如 `CCT` 和 `CJK`。Jjgod已经做了一个 `ctex` 包的 \LaTeX 接口，名为 `cxetex`，可以参见<http://www.newsmth.net/att.php?s.460.257121.452.bz2>。

在 `cxetex` 中使用 `zhspacing`，对文件 `ctexcjk.clo` 的一个轻微的修改是必需的。可以照着下面的修改相应的行（% 意味着原始行）。

```
%\XeTeXlinebreaklocale "zh"  
%\XeTeXlinebreakskip = 0pt plus 1pt minus 0.1pt  
%\setmainfont[BoldFont={cwTeXHeiBold},  
%    ItalicFont={cwTeXKaiItalic},  
%    Mapping=tex-text]{cwTeXMing}
```

⁶我曾经使用`\begingroup` 和`\endgroup` 来作为字体 `saving/restoring` 命令。然而，我发现 `ulem` 是`\hskip` 和`\penalty` 的重定义来结束当前的 `hbox` 组和开始一个新的 `hbox`，这样就会发生组的不配对。因此当在 \LaTeX 中使用 `NFSS` 信息代替组时，我改变使用字体的 `saving/restoring` 命令。但是新的 `hbox` 组中，默认是没有中文字体的，因此中文字符不能显示。因而我又在 `zhulem` 中改变 `ulem` 的新的定义，并且建议用户使用这个修改后的版本。

```
%\setsansfont[BoldFont={cwTeXHeiBold},
%    Mapping=tex-text]{cwTeXYen}
%\setmonofont{cwTeXFangSongTT}
\usepackage{zhspacing}
\newfontfamily\zhfont[BoldFont=SimHei]{SimSun}
\newfontfamily\zhcjkextafont{Sun-ExtA}
\newfontfamily\zhcjkextbfont{Sun-ExtB}
\zhspcing
```

你当然可以按照自己的口味来修改这些行。

4 获得 zhspacing

zhspacing 是在 googlecode 上的一个开源的项目。项目的主页是<http://code.google.com/p/zhspacing/>。你可以从下载页面或通过 SVN 获得最新的源代码。

文件 II

zhfont.sty

1 简介

在 L^AT_EX 中,为了简化复杂字体的设置而去使用 NFSS , zhfont 包被制作出来。它也为伪斜体、伪粗体的定义字体增加了简单的界面。

2 包选项

目前的 zhfont 仅仅有一个选项—fakebold。该选项当当前的字体为主题时, 设置中文显示为伪粗体。

3 宏支持

`\zhrmfont`

`\zhsffont`

`\zhfont`

依照当前字体家族，这些字体命令将被执行。你可以自定义自己的字体来重新定义它们。

`\setzhmainfont`

`\setzhsansfont`

`\setzhmonofont`

简单的宏可以使用`\newfontfamily`来设置`\zh{rm/sf/tt}font`

`\newfontfamilywithslant`

创建一个伪斜体的字体家族。取一个参数作为字体名称。

例子：`\newfontfamilywithslant\zhfont{SimSun}`

`\newfontfamilywithslantandbold`

创建伪斜体的字体家族和特定的另一种粗体。有两个参数分别作为字体名称和粗体名称。

例子：`\newfontfamilywithslantandbold\zhfont{SimSun}{SimHei}`

4 例子

下面都是由 `zhfont` 开启了 `fakebold` 选项生成的例子。

`\newfontfamilywithslant\zhfont{SimSun}`

中文`\textbf{测试}`。`\textit{中文\textbf{测试}}`。}

`\newfontfamilywithslant\zhfont{FangSont_GB2312}`

中文`\textbf{测试}`。`\textit{中文\textbf{测试}}`。}

中文**测试**。中文**测试**。 中文**测试**。中文**测试**。

文件 III

zhmath.sty

1 简介

zhmath 是一个允许在数学公式中显示中文字符的普通的宏包。它也允许为字母和数字改变成数学字体。

2 包选项

在 L^AT_EX 中, zhmath 有两个选项—active 和 noactive。它们是设置在数学公式中显示中文字符的一种方法。默认的是 noactive。当 active 被设置时, 数学公式中的中文字符就被激活, 当执行时, 在 `\zhmathfont` 中显示相应的字符。这种行为特别像包 mathcjk。不同样式的字号也可用 `\mathcjksizea/b/c/d` 来设置。当 noactive 被设置时, 中文字符的数学编码将会使用特殊的字体来显示。这种方式比使用 active 下的字体更加自然。active 和 noactive 都有缺点。在 active 模式下, 中文数学字体大小是由 `\mathcjksizea/b/c/d` 决定, 因此它们与正文字体大小之间总有一些独立, 也就是说当正文字号改变时, 它们的大小并不能随之改变, 还会产生一些不好的结果。然而, 在 noactive 模式下, 尽管中文数学字体大小能够随着正文字号的改变而变化, 在 `\mathrm` 等中的中文不能被显示, 因为数学字体家族被用来设置 `\mathrm` 的。因此选择你自己需要的选项。在 plain T_EX 中没有包选项可供使用。

3 宏支持

`\setzhmathfont`

设置数学字体来显示中文。有一个参数作为字体命令。仅在导言中使用。

`\zhmathfont`

该命令决定显示中文的数学字体。然而, 如果你在 L^AT_EX 中使用 noactive 选项, 或者在 plain T_EX 中, 当中文在数学公式中出现时且当 `\setzhmathfont`

被调用时，它并不会执行。因此你最好在改变后调用`\setzhmathfont`
`\zhmathfont`。

`\usecustommathfonts`

为字母和数字使用自定义的数学字体。默认是 Times New Roman。

`\setalphanummathfonts`

为显示字母和数字设置数学字体。仅在导言中使用。在 \LaTeX 中有一个参数作为字体家族，而在 plain \TeX 中有两个参数分别为字母和数字作为字体命令。

`\zhmathalphanumfont`

在 \LaTeX 中为字母和数字指定数学字体的字体命令。在它改变后应该调用`\setalphanummathfonts\zhmathalphanumfont`。

`\zhmathletterfont`

`\zhmathnumberfont`

在 plain \TeX 中为字母和数字指定数学字体的字体命令。在它们改变后`\setalphanummathfonts` 的相应的参数应该被调用。

4 例子

```
{
$中文 in math 测试$
\usecustommathfonts
\Large $能量E = mc^2$, $s_{总} = \{1 \over 2\} at^2$
}
}
```

中文 $inmath$ 测试 能量 $E = mc^2$, $s_{总} = \frac{1}{2}at^2$