

熟悉 L^AT_EX

L^AT_EX 是一种基于 T_EX 的文档排版系统。T_EX 只这么交错起伏的几个字母，便道出了“排版”二字的几分意味：精确、复杂、注重细节和品位。而 L^AT_EX 则为了减轻这种写作排版一肩挑的负担，把大片排版的格式细节隐藏在若干样式之后，以内容的逻辑结构统帅纷繁的格式，遂成为现在最流行的科技写作——尤其是数学写作的工具之一。

无论你是因为心慕 L^AT_EX 漂亮的输出结果，还是因为要写论文投稿被逼上梁山，都不得不面对一个事实：L^AT_EX 是一种并不简单的计算机语言，不能只点点鼠标就弄好一篇漂亮的文章，也不是一两个小时的泛泛了解就尽能对付得过去的^①。很得拿出点上学搞研究时候的那股钻研劲儿，才能通过手指下的键盘，编排出齐整漂亮的文章来。

L^AT_EX 的读音和写法

T_EX 一名源自 technology 的希腊词根 τεχ, T_EX 之父高德纳^②近乎固执地要求^[85] 它的发音必须是（按国际音标）[tɛx]，尽管英语中它常被读作 [tek]。（同样，高德纳教授也近乎固执地要求别人说他的姓 Knuth 时不要丢掉“K”，叫他 Ka-NOOTH，尽管在英语环境他时常会变成 Nooth 教授）。对比汉语，T_EX 的发音近似于“泰赫”，而且可以用汉语拼音准确地拼出来：tèh（或许老一辈的人习惯用注音：ㄊㄞˋㄏㄨㄛˋ）。

L^AT_EX 这个名字则是把 L^AT_EX 之父 Lamport^③ 的姓和 T_EX 混合得到的。所以 L^AT_EX 大约应该读成“拉泰赫”。不过人们仍然按着自己的理解和拼写发音习惯去读它：[ˈlɑːtɛk]、[ˈleɪtɛk] 或是 [lɑːˈtɛk]，甚至不怎么合理的 [ˈleɪtɛks]。好在 Lamport 并不介意 L^AT_EX 到底被读做什么。“读音最好由习惯决定，而不是法令。”——Lamport 如是说 [89, § 1.3]。

两个创始人对于名称和读音的不同态度或许多少说明了这样一个事实：L^AT_EX 相对原始的 T_EX 更少关注排版的细节，因此 L^AT_EX 在很多时候并不充当专业排版软件的角色，而只是一个文档编写工具。而当人们在 L^AT_EX 中也保以追求完美的态度并用到一些平时不大使用的命令时，通常总说这是在 T_EX 层面排版——尽管 L^AT_EX 本身正是运行于 T_EX 之上的。

类似地，T_EX 和 L^AT_EX 字母错位的排印也体现出一种面向排版的专业态度，即使在字符难以错位的场合，也应该按大小写交错写成 TeX 和 LaTeX。

现在我们使用的 L^AT_EX 格式版本为 2_ε，意思是超出了第 2 版，接近却没有达到第 3

^① 是的，有一个著名的入门教程就叫《112 分钟学会 L^AT_EX》[123]。不过这个分钟其实是以页码计算的，粗略浏览一遍还远算不上学会；而且即使掌握了这个教程中的内容，仍然可能在实际写作中遇到许多难以解决的问题。本书同样不打算让你能迅速变成一个高手。

^② Donald Ervin Knuth, Stanford 大学计算机程序设计艺术荣誉教授，Turing 奖和 von Neumann 奖得主。高德纳是他的中文名字。T_EX 系统就是高德纳为了排版他的七卷本著作《计算机程序设计艺术》而编制的。

^③ Leslie Lamport 博士，微软研究院资深研究员，Dijkstra 奖得主。

是 Hartke [70], 里面介绍了 20 多种带有数学支持的免费 $\text{T}_\text{E}\text{X}$ 字体。该文档在 $\text{T}_\text{E}\text{X}$ Live 和 $\text{C}_\text{T}_\text{E}\text{X}$ 套装中都有收录, 是使用成套字体的很好的参考。一个收录更为全面的 $\text{L}_\text{T}_\text{E}\text{X}$ 字体目录是 Jørgensen [80], 它展示了 $\text{T}_\text{E}\text{X}$ Live 2006–2007 中所能使用的 182 种字体。

下面来看现代的方法, 使用 $\text{X}_\text{T}_\text{E}\text{X}$ 来选择字体。在 $\text{X}_\text{T}_\text{E}\text{X}$ 中, 主要使用 `fontspec` 宏包的机制来调用字体。最基本的是设置正文罗马字体族、无衬线字体族和打字机字体族的命令:

```
\setmainfont[<可选选项>]{<字体名>}
\setsansfont[<可选选项>]{<字体名>}
\setmonofont[<可选选项>]{<字体名>}
(可用的可选选项参看 fontspec 文档 Robertson and Hosny [144])
```

例如:

```
% 在导言区设置全文字体为 Windows 提供的 Times New Roman, Verdana, Courier New
\usepackage{fontspec}
\setmainfont{Times New Roman}
\setsansfont{Verdana}
\setmonofont{Courier New}
```

2-1-37

此时 `\rmfamily`, `\sffamily` 和 `\ttfamily` 就分别对应设置的三种字体, 而且 `fontspec` 会自动找到并匹配对应的粗体、斜体等变体, 尽量使 `\bfseries`, `\itshape` 等命令也有效。

也可以定义新的字体族命令:

```
\newfontfamily<命令>[<可选选项>]{<字体名>}
```

例如我们为 Java 运行库附带的 Lucida Sans 字体定义一个命令 `\lucidasans`:

```
% 导言区使用
\newfontfamily\lucidasans{Lucida Sans}
% 正文使用
{\lucidasans This is Lucida Sans.}
```

2-1-38

This is Lucida Sans.

$\text{X}_\text{T}_\text{E}\text{X}$ 下中文字体的设置使用 `xeCJK` 宏包^[193] (`ctex` 宏包或文档类会自动调用它)。`xeCJK` 提供了与 `fontspec` 对应的中文字体设置命令:

```
\setCJKmainfont[<可选选项>]{<字体名>}
\setCJKsansfont[<可选选项>]{<字体名>}
\setCJKmonofont[<可选选项>]{<字体名>}
\setCJKfamilyfont{<中文字体族>}[<可选选项>]{<字体名>}
```

这些命令对于 Linux 等系统下的中文用户特别有用, 因为 `ctex` 宏包默认是针对 Windows 的字体配置的, 可以用 `nofonts` 选项禁用预定义的中文字体设置, 而自己定义字体。例如使用文鼎公司免费发布的字体:

```
% 在导言区设置全文字体为“文鼎 P L 报宋二 GBK”
% 并设置中文的 kai 字体族为“文鼎 P L 简中楷”
\setCJKmainfont{AR PLBaosong2GBK Light}
\setCJKfamilyfont{kai}{AR PL KaitiM GB}
```

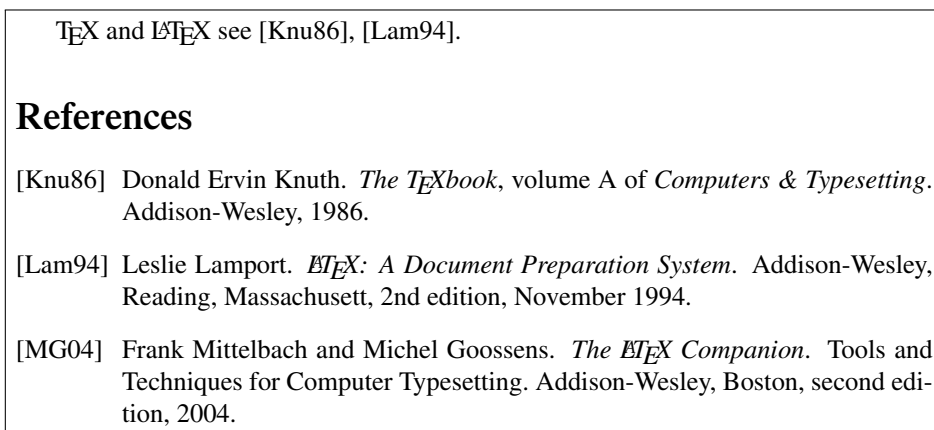


图 3.6: alpha 格式的引用和文献列表



图 3.7: TeXworks 与 WinEdt 中的 BibTeX 命令按钮

如图 3.8 所示, BibTeX 的参考文献列表和引用也是通过读写一系列辅助文件完成的, 不过这个功能比一般的目录和交叉引用生成还要复杂一些。要得到完整的文献列表和正确的引用信息, 至少需要四个步骤:

1. 编译 .tex 源文件, 生成没有文献列表的 PDF 文件, 同时将 `\cite` 命令产生的引用信息、`\bibliography` 指定的数据库名、`\bibliographystyle` 指定的文献格式名都写入 .aux 辅助文件;
2. 使用 bibtex 程序处理第一次编译得到的 .aux 辅助文件, 按照其中记录的引用、文献信息, 从 .bib 数据库中提取出排版参考文献列表的 L^ATeX 代码, 写入 .bbl 文件;
3. 再次编译 .tex 源文件, 读入上一步生成的 .bbl 文件, 生成有文献列表的 PDF 文件, 同时将 `\cite` 的引用信息再次写入 .aux 辅助文件;
4. 第三次编译 .tex 源文件, 读入前面生成的 .bbl 文件, 在指定位置生成文献列表, 读入上一步生成的 .aux 辅助文件, 在引用处生成正确的引用编号信息, 得到有正确文献列表和引用的 PDF 文件。

上面的过程中, 后面生成 `\cite` 命令引用编号的信息与一般的交叉引用没有区别, 只是前面增加了调用 BibTeX 生成文献列表的过程。当然, 这一流程是针对最后输出完全正确的文献、引用信息而言的, 在文档编写、修改过程中, 如果不是为了检查最终的结果, 只需要编译一遍就可以了。

*本节后面内容初次阅读可跳过

带上下限的数学算子使用起来与巨算符相似^①：

```
\begin{equation}
\varlimsup_{k\to\infty} A_k = \lim_{J\to\infty} \lim_{K\to\infty}
\bigcap_{j=1}^J \bigcup_{k=j}^K A_k
\end{equation}
```

4-3-7

$$\overline{\lim}_{k\rightarrow\infty} A_k = \lim_{J\rightarrow\infty} \lim_{K\rightarrow\infty} \bigcap_{j=1}^J \bigcup_{k=j}^K A_k \quad (4.4)$$

尽管 L^AT_EX 已经预定义了许多算子名，实际中仍不免捉襟见肘。因而 amsmath 提供了 \DeclareMathOperator 命令来声明新的算子名，其用法与 \newcommand 相似。如果使用带星号的 \DeclareMathOperator* 命令，则可以声明带上下限的数学算子。这两个命令都只用于导言区：

```
% 导言区 \usepackage{amsmath}
\DeclareMathOperator{\card}{card} % 集合基数
\DeclareMathOperator*{\esssup}{ess\,sup} % 本性上确界
```

4-3-8

如果只是在一两个公式中临时使用，也可以不专门定义命令，而是使用 \operatorname{<内容>} 来表示；带星的 \operatorname* 则是上下限形式的上下标。例如：

```
\[ \operatorname*{Prob}_{\{1,\ldots,n\}}
(\text{不可能事件}) =
\operatorname{card}(\varnothing)/n = 0. \]
```

$$\operatorname*{Prob}_{\{1,\dots,n\}}(\text{不可能事件}) = \operatorname{card}(\emptyset)/n = 0.$$

4-3-9

可以使用 \mathop 命令来让 T_EX 将其参数看做是数学算子。例如，也可以使用 $\mathop{\operatorname{sin}} x$ 来代替 $\sin x$ 得到 $\sin x$ ，实际上 \operatorname 等命令的定义就离不开 \mathop。下面这个例子来解释如何综合使用 \mathord 和 \mathop 连续改变符号的类型，实现 \sideset 的功能：

```
% 用原始命令实现 \sideset 的功能
\[ \mathop{\mathord{\sum}}'
\limits_{i=1}^n A_n \]
```

$$\sum'_{i=1}^n A_n$$

\mathop 有一个有用的功能，就是用它来改进单字符数学算子的排版。前面我们在排版积分微元 dx 时，经常需要在前面手工加上 \，表示间距，但如果要排版 $\frac{dx}{dt}$ 这种分式，又不应该加上间距了。如果要给微分算子 d 定义一个命令 \diff，能否让它在不同的情况下都产生正确的间距呢？下面的定义解决了这个问题：

```
\newcommand\diff[1]{\mathop{\operatorname{d}}#1}
\[
\int_0^1 \int_0^1 f(x,y) \int_0^1 \frac{\diff z}{g(x,y,z)} \diff x \diff y
\]
```

4-3-10

$$\int_0^1 \int_0^1 f(x,y) \int_0^1 \frac{dz}{g(x,y,z)} dx dy$$

^① 这里 \to 就是右箭头符号 → 的命令，见 4.3.3 节。

```

\begin{array}{@{}ccc:c@{}}
  a_{11} & a_{12} & a_{13} & b_1 \\
  a_{21} & a_{22} & a_{23} & b_2 \\
  a_{31} & a_{32} & a_{33} & b_3 \\
  \cdashline{1-3}
  0 & 0 & 0 & b_4 \\
\end{array}
\right)
\]

```

5-1-41

$$\left(\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \\ \hline 0 & 0 & 0 & b_4 \end{array} \right)$$

此外，如果同时使用了 `array` 宏包，则还可以使用 `\firsthline` 与 `\lasthline` 的虚线对应命令 `\firsthdashline` 与 `\lasthdashline`。

`arydshln` 宏包中虚线的格式可以使用 `\dashlinedash` 和 `\dashlinegap` 两个长度变量控制。它们分别表示虚线的黑白长度，默认值都是 4pt。可以修改它们得到特别的虚线，例如改为密集的点线风格：

```

% \usepackage{arydshln}
\setlength\dashlinedash{1pt}
\setlength\dashlinegap{2pt}
\begin{tabular}{:cc:cc:}
\hdashline
上 & 上 & 上 & 上 \\
\cdashline{1-2}
下 & 下 & 下 & 下 \\
\hdashline
\end{tabular}

```

5-1-42

```

上 上 上 上
下 下 下 下

```

进一步地，命令 `\hdashline` 和 `\cdashline` 的后面也可以使用 `[{线}/空]` 的格式设置某一条虚线的黑白长度；竖线则可以使用带一个参数的列格式说明符 `{(线)/空}`。例如：

```

% \usepackage{arydshln}
\begin{tabular}{;{8pt/2pt}cc;{2pt/2pt}cc;{8pt/2pt}}
\hdashline[8pt/2pt]
上 & 上 & 上 & 上 \\
\cdashline{1-2}[2pt/2pt]
下 & 下 & 下 & 下 \\
\hdashline[8pt/2pt]
\end{tabular}

```

5-1-43

TeX 引擎命令	图形驱动	支持的格式	备注
latex	Dvips	EPS	MiKTeX 还部分支持 PNG 和 JPEG
latex	DVIPDFMx	EPS, PDF, PNG, JPEG	PDF、PNG、JPEG 需要使用 extractbb 程序生成 .xbb 文件
pdflatex		MPS, PDF, PNG, JPEG	MPS 是 METAPOST 的输出格式, TeX Live 2010 以后还能自动将 EPS 转换为 PDF 文件插入
xelatex	xdvipdmtx	EPS, PDF, PNG, JPEG, BMP	驱动是自动调用的, MAC 系统下的旧驱动 xdv2pdf 还支持其他一些格式

表 5.5: L^AT_EX 编译程序与插图格式

```
% 导言区 \usepackage{graphics}
% 或 \usepackage{graphicx}
狮子:\includegraphics{lion}
```



5-2-1

graphics 宏包与 graphicx 宏包在功能上并没有什么差别。graphicx 宏包支持 (项目)=(值) 的语法, 使用起来更为方便, 因此本节下面的介绍都以 graphicx 宏包为准。以下总假定文档中已经使用了

```
\usepackage{graphicx}
```

插图的核心命令是 \includegraphics, 其语法是:

```
\includegraphics[<选项>]{<文件名>}
```

其中 <文件名> 是图形文件的文件名, 一般文件的后缀可以省略不写, L^AT_EX 会自动查找它支持的文件格式, 为了明确也可以加上后缀。

插入的图形都有一个自然比例, 对于 EPS、PDF 图形就是制做的尺寸; 对于 JPG、PNG、BMP 等像素图的尺寸则是点阵数除以图形打印度 (一般用每英寸点数 DPI 表示)。可以给 \includegraphics 命令加一些可选项来调整图形的大小、位置等。可以用命令选项 width、height 和 scale 设置图形的宽度、高度或缩放比例, 这也是最常用的几个插图选项。例如:

```
\includegraphics[width=2em]{lion}
\includegraphics[height=1cm]{lion}
\includegraphics[scale=0.5]{lion}
```



5-2-2

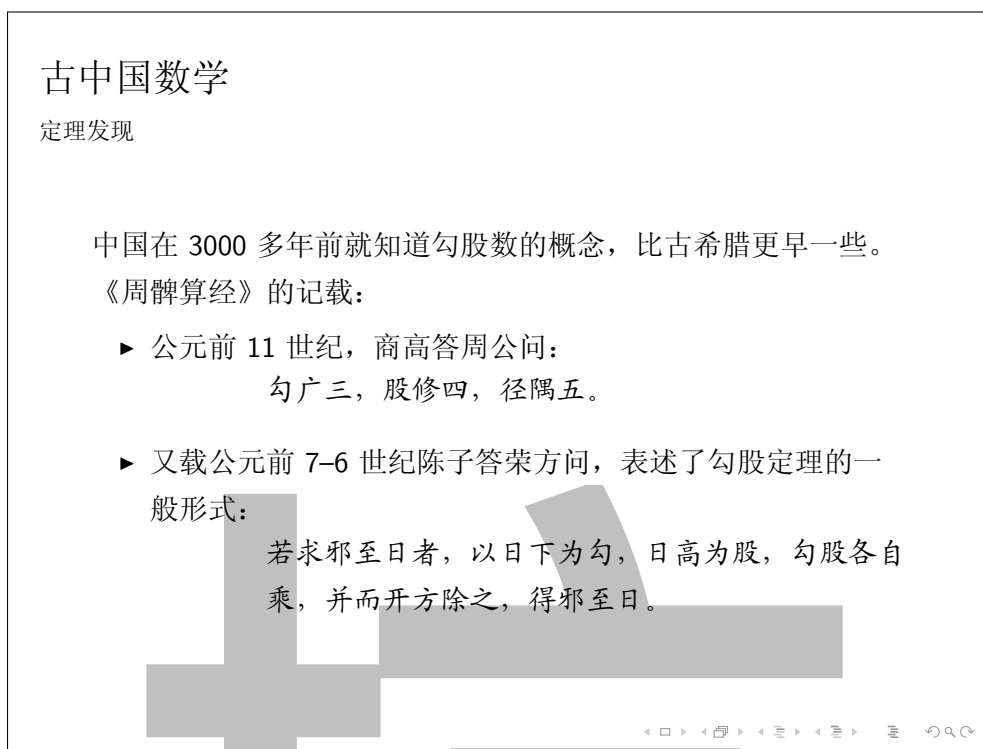


图 6.1: 基本的 frame 环境

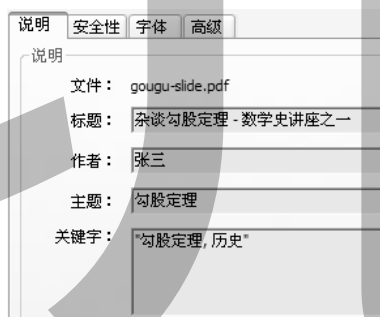


图 6.2: 在 Adobe Reader 中查看 beamer 生成的 PDF 文件说明信息

`\tableofcontents` 命令也必须放在帧里显示。因而我们的幻灯片就要这样写：

```
\begin{frame}{目录}
  \tableofcontents
\end{frame}

\section{勾股定理在古代}
...
```

`\part` 命令本身也不产生标题。但 `beamer` 提供了一个 `\partpage` 命令，它与 `\titlepage` 类似，可以在一帧中产生文档某部分的标题。例如：

```
\part{引言}
\begin{frame}
  \partpage
```

☞ **Label(s) may have changed. Rerun to get cross-references right.**

这个警告出现在编译的最后，说明经过编译引用标签的位置可能发生变化，可以通过重新编译保证不发生这种问题。如果不是最终输出，可以忽略这个问题。

☞ **Marginpar on page ... moved.**

边注被移动，不在所标注文字的位置。这一般发生在多条边注相距很近或是边注在一页的顶部或底部时。这一问题通常可以忽略。

☞ **No \author given.**

使用了 `\maketitle` 却没有使用 `\author` 指定作者。

☞ **Optional argument of \twocolumn too tall on page ...**

双栏命令的可选参数（单栏的内容）超出一页高度，造成问题。使用 `multicol` 宏包来实现双栏是解决这个问题的一种方法（2.4.4 节）。

☞ **Reference ‘...’ on page ... undefined.**

引用的标签未定义。如果这出现在第一次编译中，那么是辅助文件尚未生成造成的正常现象；如果多次编译仍有问题，一般是拼写错误。

☞ **Some font shapes were not available, defaults substituted.**

如前所述，如果文档中有某种字体形状不存在，那么在文档末尾也会出现这条警告。如果没有造成效果偏差，可以忽略。

☞ **There were multiply-defined labels.**

如前所述，如果文档中有重复定义的标签，就会在文档末尾出现这条警告。这实际上是一个错误，应该修改。

☞ **There were undefined references or citations.**

如前所述，如果文档中有未定义的交叉引用或文献，就会在文档末尾出现这条警告。

☞ **Unused global option(s): [...]**

文档类中使用了全局的选项，但这个选项没有在任何宏包中发挥作用。一般删掉多余的全局选项即可。

7.2 调试与分析




单单了解 \LaTeX 的错误信息有时并不能完全解决问题。隐藏的错误可能并不会让 \LaTeX 报错，但会造成错误的结果；复杂的错误即使产生报错信息也可能让人很难理解。说到底，排查错误还需要作者本人的理解和努力。下面我们就来看看对 \LaTeX 文件作进一步调试分析的方法。

7.2.1 调试命令

调试分析的首要任务是定位造成问题的代码。对于简单错误这只需要查看第一个报错信息的文件名和行号，或者在输出结果中寻找不对的地方。但还有一些来源更隐蔽的问题，比如命令重定义、宏包冲突、命令的错误定义等，显示出错误的地方就往往不是产生错误的源头了。这个时候可以使用排除法，逐步去掉不影响错误的内容，最终把错误定位在一小段代码上。如果采用二分法，每次排除掉一半内容，那么最多试验几十次，总可以找到出问题的地方。

*本节内容初次阅读可略过。

 \TeX 中字符的意义并非一成不变，每个字符的都有一个类别，表示它是转义符（通常的 \backslash ）、分组的括号、数学环境、表格对齐、参数、上下标、空格、字母、其他、活动符（通常的 \sim ）、注释符等等 [85, Chapter 7]。类别可以使用 `\catcode` 命令来设置。利用这种方式，可以完全改变 \TeX 的语法。像 `verb` 这类命令就是通过修改字符平常的类别完成的。在 \LaTeX 中，最常用的是把字符定义为像 \sim 一样的活动符，然后把字符直接定义为命令。例如，我们用双引号表示加粗命令：

8-1-13

```
\catcode'\="=\active
\def"#1"{\textbf{#1}}
An "important" usage.
```

An important usage.

这里 `'\"` 就是字符 `"` 的编码 34（不同的数字表示法见 2.1.2 节），而 `\active` 则是活动符的类型编号 13。

在 \LaTeX 中有关字符类别的一种常见用法是将字符 `@` 在字母类别和其他类别之间转换。当 `@` 是字母时就可以在命令名中使用 `@`，相当于一个不同于正常文档的名字空间，因而在 \LaTeX 的内部命令及宏包编写时经常会遇到。可以用 `\makeatletter` 和 `\makeatother` 命令分别让 `@` 成为字母或特殊符号，这经常用于修改内部命令的定义。例如^[137]：

8-1-14

```
\makeatletter
\setlength\@fpsep{1cm}% 浮动页中不同浮动体间距
\makeatother
```

练习

8.1 重定义 `\cite` 命令，让引用的编号或文字使用粗体显示（一般的引用格式设置参考 3.3.3 节）。

8.2 绘图宏包 `PSTricks` 和 `TikZ` 都设计了更为强大的循环命令，前者是 `multido` 宏包的 `\multido`，后者是 `pgffor` 宏包的 `\foreach`。查看文档 `Tantau` [163]；`Zandt` [187]，看看它们有什么特别之处。

8.1.2 编写自己的宏包和文档类



如果你已经熟悉了 \TeX 宏和 \LaTeX 的各种功能，那么编写自己的宏包也就水到渠成了。所谓宏包就是把许多宏定义汇集在一起的结果，最简单的宏包就是一个简单的 \TeX 源文件，里面是你所写的宏定义：

```
% 最简单的宏包
% boldemph.tex
\newcommand\Emph[1]{\textbf{#1}}
```

使用这种简单的宏包也很简单，只要在正文中使用 `\input{boldemph}` 就可以了。

不过，这种定义方式有许多缺点，比如如果多次使用 `\input` 引用同样的文件，那么就会产生命令、计数器、盒子等变量重定义的错误，代码中如果有计算还可能发生其他隐藏的错误。这样的简单文件功能也很简单，比如我们不能给宏包增加选项，不能向宏包代码中所引用的宏包传递参数等等。

*本节内容初次阅读可略过。

- [166] THE L^AT_EX3 PROJECT. *L^AT_EX 2_ε for class and package writers*, 2006
CTAN:/macros/latex/doc/clsguide.pdf
- [167] KRESTEN KRAB THORUP, FRANK JENSEN, and CHRIS ROWLEY. *The calc package—Infix notation arithmetic in L^AT_EX*, 2007
CTAN://macros/latex/required/tools/calc.pdf
- [168] HÀN THẾ THÀNH, SEBASTIAN RAHTZ, HANS HAGEN, HARTMUT HENKEL, PAWEŁ JACKOWSKI, and MARTIN SCHRÖDER. *The pdfT_EX user manual*, version 1.675, 2007
CTAN://systems/pdftex/pdftex-a.pdf
- [169] THOMAS TITZ. *The idxlayout package*, version 0.4c, 2010
CTAN://macros/latex/contrib/idxlayout/idxlayout.pdf
- [170] MARK TRETTIN. *An essential guide to L^AT_EX 2_ε usage—Obsolete commands and packages*, version 1.8.5.7, 2007. English translation by Jürgen Fenn
CTAN://info/l2tabu/english/l2tabuen.pdf
- [171] TUG WORKING GROUP ON A T_EX DIRECTORY STRUCTURE (TWG-TDS). *A Directory Structure for T_EX Files*, version 1.1, 2004
CTAN://tds/
- [172] UK TUG. *The UK T_EX FAQ: Your 444 Questions Answered*, version 3.20, 2010
CTAN://help/uk-tex-faq/newfaq.pdf
- [173] HIDEO UMEKI. *The geometry package*, version 5.3, 2010
CTAN://macros/latex/contrib/geometry/geometry.pdf
- [174] VLADIMIR VOLOVICH, WERNER LEMBERG, and L^AT_EX3 PROJECT TEAM. *Cyrillic languages support in L^AT_EX*, 1999
CTAN://macros/latex/doc/cyrguide.pdf
- [175] HERBERT VOSS. *Math mode*, version 2.43, 2009
CTAN://info/math/voss/mathmode/Mathmode.pdf
- [176] STRUNK WILLIAM, JR. and EDWARD A. TENNEY. *The Elements of Style*. Boston et al: Allyn and Bacon, fourth edition, 2000
- [177] PETER WILSON. *The hyphenat package*. Herries Press, version 2.3b, 2004
CTAN://macros/latex/contrib/hyphenat/hyphenat.pdf
- [178] PETER WILSON. *The layouts package: User manual*. Herries Press, version 2.6d, 2004
CTAN://macros/latex/contrib/layouts/layman.pdf
- [179] PETER WILSON. *The changepage and chngpage packages*. Herries Press, version 1.2b and 1.0c, 2009
CTAN://macros/latex/contrib/changepage/changepage.pdf
- [180] PETER WILSON. *The chngcntr package*, version 1.0a, 2009
CTAN://macros/latex/contrib/chngcntr/chngcntr.pdf
- [181] PETER WILSON. *The fonttable package*. Herries Press, version 1.6, 2009
CTAN://macros/latex/contrib/fonttable/fonttable.pdf
- [182] PETER WILSON. *The titling package*. Herries Press, version 2.1d, 2009
CTAN://macros/latex/contrib/titling/titling.pdf
- [183] PETER WILSON. *The tocbibind package*. Herries Press, version 1.5j, 2009
CTAN://macros/latex/contrib/tocbibind/tocbibind.pdf
- [184] PETER WILSON. *The tocloft package*. Herries Press, version 2.3d, 2009