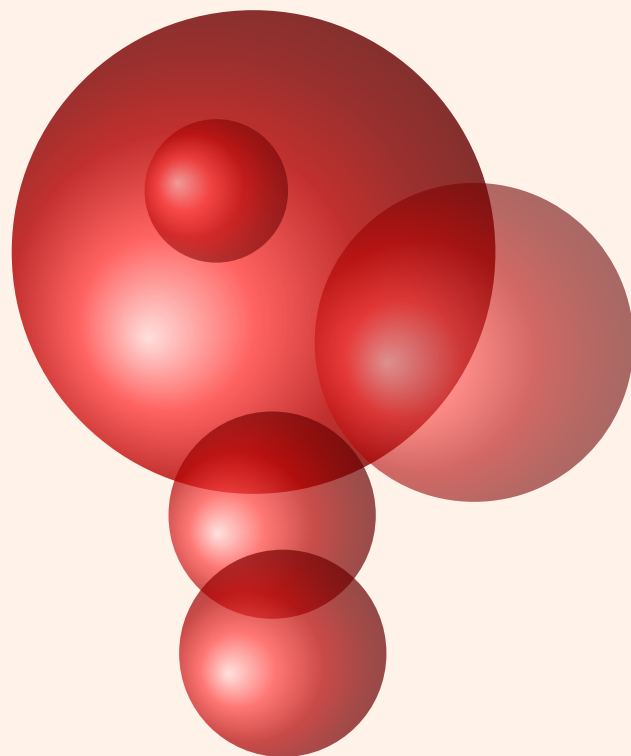
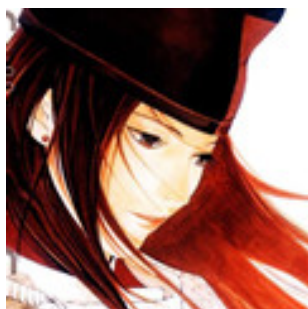


L^AT_EX 札记 1.0



Joseph Pan

<http://josephpan.net>



Joseph Pan

Joseph Pan L^AT_EX 札记

这是一份针对初学者而编写的 L^AT_EX 札记。我写这份文档的主要目的是介绍一下 L^AT_EX 的优点，一些小技巧，还有针对初学者常犯的几个错误，提醒一下他们需要格外引起注意的问题。

重点将讲讲中文处理，以及使用 *JabRef* 管理 *BibTeX* 的方法，这是很多 L^AT_EX 教程所缺少的内容。

因此，这份材料不是一份面面俱到的教程，而适合当作其他教程的补充。其他的内容可以参考 *lshort* 或其他教程，里头有更丰富的介绍。

Anyway，希望这份东西对正在学习 L^AT_EX 的人有用。☺

Happy TeXing!



[L^AT_EX 札记 July 23, 2012]

☞ 感谢 [Alain Matthens](#) 写了 `tkz-doc`，这份文档的风格修改自 `tkzdoc.cls`。

☞ 感谢 [CTeX 网站](#)、[ChinaTeX 网站](#) 等相关网站以及 [李树钧](#)、[王垠](#) 等提供的相关资料。

☞ 文中用到了一些 [棋魂](#) 的漫画图案，纯粹出于我对该漫画的喜爱，棋魂的版权归日本《少年 JUMP 周刊》（集英社）所有。

Please report typos or any other comments to this documentation to [Joseph Pan](#) This file can be redistributed and/or modified under the terms of the L^AT_EX Project Public License Distributed from CTAN archives in directory `CTAN://macros/latex/base/lppl.txt`.



Contents

1	漫谈 (La)TeX	5
1.1	TeX 的诞生与发展	5
1.2	TeX 趣事	5
1.2.1	TeX 与 π	5
1.2.2	TeX 支票	5
1.2.3	冰冻 TeX	6
1.3	TeX 和 L ^A T _E X	6
1.4	TeX 系统	7
1.5	TeX 除了编辑论文还能做什么?	7
1.5.1	L ^A T _E X 绘图	8
1.5.2	L ^A T _E X 吉他谱	9
1.5.3	L ^A T _E X 的文字特效	10
2	最简单的文档示例	11
2.1	分析	11
2.1.1	文档类型	11
2.1.2	引言 (preamble)	12
3	几个小技巧	13
3.1	获取帮助	13
3.2	编译当前一小段	13
3.3	清理辅助文件	13
3.4	注释	13
3.4.1	单行注释 (%)	13
3.4.2	多行注释	14
3.5	英文断字的处理	14
3.5.1	使用命令 <code>\hyphenation</code>	14
3.5.2	使用盒子	15
3.6	断行和分页	15
3.7	插图技巧	15
3.7.1	插图方法	15
3.7.2	多个图形	16
3.7.3	旋转和缩放图案	17
3.7.4	给图像加边框	18
4	初学者要注意的地方	21
4.1	特殊字符	21
4.2	标点符号	21
4.2.1	引号	21
4.2.2	省略号	22
4.2.3	破折号	22
4.2.4	波浪号 (~)	23
4.2.5	度的符号 (°)	23
4.2.6	数学符号	23
5	中文处理	24

6 参考文献	28
6.1 JabRef 简介	28
6.2 利用 Google Scholar 获取 BibTeX	28
6.3 在 L ^A T _E X 中引用文献	29
Index	31

1 漫谈 (La)TeX

1.1 TeX 的诞生与发展

七十年代末, Donald E. Knuth (高德纳) 在看到其多卷巨著 “The Art of Computer Programming” 第二卷的校样时, 对由计算机排版的校样的低质量感到无法忍受。因此决定自己来开发一个高质量的计算机排版系统, 这样就有了 TeX。



Figure 1: This is TeX!

TeX 的源程序是用 Pascal 写成的, 原因是 Knuth 希望 TeX 尽可能方便地移植到其它的操作系统中去。当时 Pascal 是最适合于这一要求的编程语言。这也使得 TeX 现在已经在几乎所有的计算机系统中得到实现。

TeX 的另一个重要的特征就是它的输出是与设备无关的。TeX 的输出文件称为 DVI 文件, 即是 “Device Independent”。一旦 TeX 处理了你的文件, 你所得到的 DVI 文件就可以被送到任何输出设备如打印机, 屏幕等并且总会得到相同的结果, 而这与这些输出设备的限制没有任何关系。这说明 DVI 文件中所有的元素, 从页面设置到文本中字符的位置都被固定, 不能更改。

1.2 TeX 趣事

1.2.1 TeX 与 π

TeX 的第一版于 1978 年面世。在经过了不断的改进后, 1982 年版的 TeX 是一个十分稳定的版本。从此以后, TeX 没有较大的变动, 只有很少部分的改进和错误修正。现在几乎所有的操作系统平台下, 都有相应的 TeX 软件, 而且用它们排版同一个文件得到的输出结果是相同的。TeX 目前的版本是 3.1415926, 很接近圆周率 π 。每有一新的版本, 就会加上一个 π 的小数点后的数。所以我们已经知道下一版本的 TeX 是 3.14159265, 最终将收敛到 π 。这也代表了 TeX 不断追求完美的理想。

1.2.2 TeX 支票

Knuth 用他自己设计的 “文学编程语言” 写了 TeX 的代码, 到了正式发行的时候, 他非常自信的奖励 \$ 1.28 给第一个发现 bug 的人, 以后每发现一个 bug, 奖金都会翻倍, 最多 \$ 327.68。后来由

于加入了对 8 字节字符的支持，产生了一些小 bug。从 1995 年到现在，总共只发现一个 bug。发现 bug 的周期越来越长，而且后来的 bug 都是一些在挑剔的人找出来的一些“美学”上的问题了。但是严谨的 Knuth 仍然把它们作为 bug 处理。国内的王垠就曾获得他的两张支票。

1.2.3 冰冻 T_EX

T_EX 现在已经被它的开发者 Knuth 所“冰封” (frozen)，新版本的 T_EX 只有少许改进和错误修正。这些变动必须保证不会使得即使是写于 1982 年的 T_EX 文档无法编译，T_EX 自带了测试文件来确保这一点。如果测试文件文件的输出结果不同于预定的结果，那么这个排版系统就不能够被称为“T_EX”。所以像一些对 T_EX 有很大改进的软件，如“Omega”等将不能被称为“T_EX”，尽管它也是基于“T_EX”的。

虽然 T_EX 在过去的二十多年中没有大的变化，但这并不意味着 T_EX 是一个僵化的系统。恰恰相反，T_EX 开放的设计使得它能够很容易的适应新的要求。

1.3 T_EX 和 L^AT_EX

L^AT_EX 是一个宏包集，其目的是使作者能够利用一个预先定义好的专业页面设置，从而得以高质量地排版和打印他们的作品。它与 T_EX 可以说是高层和底层的关系。相关的几个名词有：

1. T_EX

最基本的 T_EX 程序只是由一些很原始的命令组成，它们可以完成简单的排版操作和程序设计功能。然而，T_EX 也允许用这些原始命令定义一些更复杂的高级命令。这样就可以利用低级的块结构，形成一个用户界面相当友好的环境。

在处理器运行期间，该程序首先读取所谓的格式文件，其中包含各种以原始语言写成的高级命令，也包含分割单词的连字号安排模式。接着处理程序就处理源文件，其中包含要处理的真正文本，以及在格式文件中已定义了了的格式命令。

创建新格式是一件需要由具有丰富知识程序员来做的事情。把定义写到一个源文件中，这个文件接着被一个名叫 `initex` 的特殊版本的 T_EX 程序处理。它采用一种紧凑的方式存贮这些新格式，这样就可以被通常 T_EX 程序很快地读取。

2. Plain T_EX

Knuth 设计了一个名叫 Plain T_EX 的基本格式，以与低层次的原始 T_EX 呼应。这种格式是用 T_EX 处理文本时相当基本的部分，以致于我们有时都分不清到底哪条指令是真正的处理程序 T_EX 的原始命令，哪条是 Plain T_EX 格式的。大多数声称只使用 T_EX 的人，实际上指的是只用 Plain T_EX。

Plain T_EX 也是其它格式的基础，这进一步加深了很多人认为 T_EX 和 Plain T_EX 是同一事物的印象。

3. L^AT_EX

Plain T_EX 的重点还只是在于如何排版的层次上，而不是从一位作者的观点出发。对它的深层功能的进一步发掘，需要相当丰富的编程技巧。因此它的应用就局限于高级排版和程序设计人员。

Leslie Lamport 开发的 L^AT_EX 是当今世界上最流行和使用最为广泛的 T_EX 格式。它构筑在 Plain T_EX 的基础之上，并加进了很多的功能以使得使用者可以更为方便的利用 T_EX 的强大功能。使用 L^AT_EX 基本上不需要使用者自己设计命令和宏等，因为 L^AT_EX 已经替你做好了。因此，即使使用者并不是很了解 T_EX，也可以在短短的时间内生成高质量的文档。对于生成复杂的数学公式，L^AT_EX 表现的更为出色。

4. L^AT_EX 2e

L^AT_EX 自从二十世纪八十年代初问世以来，也在不断的发展。最初的正式版本为 2.09，在经过几年的发展之后，许多新的功能，机制被引入到 L^AT_EX 中。在享受这些新功能带来的便利的同时，它所伴随的副作用也开始显现，这就是不兼容性。标准的 L^AT_EX 2.09，引入了“新字体选择框架” (NFSS) 的 L^AT_EX，SLiTeX，AMSLaTeX 等等，相互之间并不兼容。这给使用者和维护者都带来很大的麻烦。

为结束这中糟糕的状况，Frank Mittelbach 等人成立了 L^AT_EX 3 项目小组，目标是建立一个最优的，有效的，统一的，标准的命令集合。即得到 L^AT_EX 的一个新版本 3。这是一个长期目标，向这个目标迈出第一步就是在 1994 年发布的 L^AT_EX 2e。L^AT_EX 2e 采用了 NFSS 作为标准，加入了很多新的功能，同时还兼容旧的 L^AT_EX 2.09。L^AT_EX 2e 每 6 个月更新一次，修正发现的错误并加入一些新的功能。在 L^AT_EX 3 最终完成之前，L^AT_EX 2e 将是标准的 L^AT_EX 版本。

1.4 T_EX 系统

T_EX 系统是集成了编译 (La)TeX 文档所需的软件和宏包的系统。常用的包括：

1. MikTeX。MiKTeX 是 Windows 下最好用的 T_EX 系统。网址：<http://www.miktex.org> 或 <http://miktex.sourceforge.net>
2. TeXLive。TeXLive 是由国际 T_EX 用户组织 TUG 开发的 T_EX 系统，支持不同的操作系统平台。其 Windows 版本又称 fpTeX，Unix/Linux 版本即著名的 teTeX。网址：<http://tug.org/texlive/>
3. CTeX。CTeX 中文套装是基于 Windows 下的 MiKTeX 系统，集成了编辑器 WinEdt 和 PostScript 处理软件 Ghostscript 和 GSview 等主要工具。CTeX 中文套装在 MiKTeX 的基础上增加了对中文的完整支持。CTeX 中文套装支持 CJK, xeCJK, CCT, TY 等多种中文 T_EX 处理方式。网址：<http://www.ctex.org>

1.5 T_EX 除了编辑论文还能做什么？

T_EX 是一个非常多才多艺的程序。它不但可以编辑论文，书籍，幻灯片，学术杂志，个人简历，还可以编辑乐谱，化学分子图，电路图，国际象棋，中国象棋，甚至围棋棋谱，……事实上只有少量文档不适合用 T_EX 编辑。

1.5.1 L^AT_EX 绘图

L^AT_EX 可以和很多绘图工具结合，精确地绘制出美妙的图形。你可以看看一些绘图工具的官方 gallery:

1. TikZ: <http://www.texample.net/>;
2. Asymptote: <http://asymptote.sourceforge.net/gallery/>;
3. MetaPost: <http://tex.loria.fr/prod-graph/zoonekynd/metapost/metapost.html>;
4. gnuplot: <http://www.gnuplot.info/screenshots/index.html#demos>

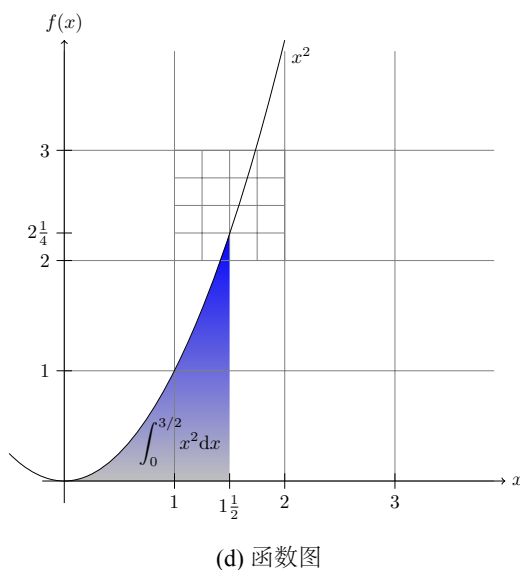
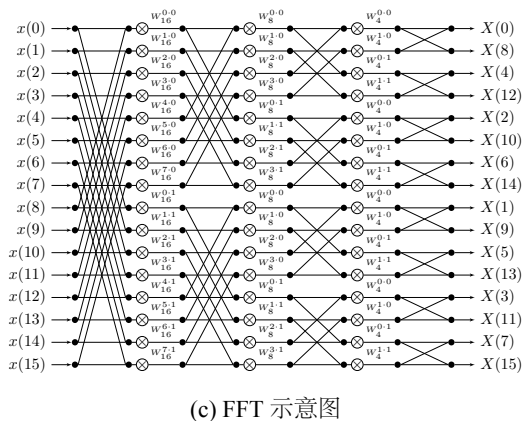
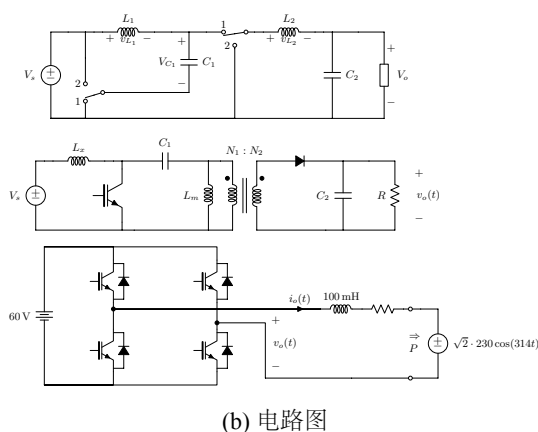
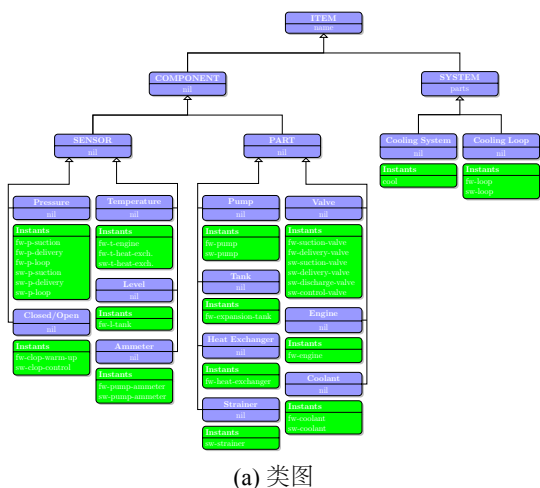


Figure 2: L^AT_EX 绘图

1.5.2 L^AT_EX 吉他谱

L^AT_EX 排版的吉他谱：

Hoquetus David (Guillaume de Machaut)

Hoquetus David

Guillaume de Machaut
(c.1300 - 1377)

Edited by M.A.B. Soloists
Transcribed for MusiX_TE_X by Moriwaki Michio

David Triplum

David Hoquetus

David Tenor

A1 a1

6

12

A2

Copyright © 2000, 2002 M.A.B. Soloists

Ver. 1.1

Figure 3: L^AT_EX 排版的吉他谱

1.5.3 L^AT_EX 的文字特效

L^AT_EX 可以生成很有趣的文字特效，如图4所示。

if i had to live my life
 without you near me the days would all be
 empty the nights would seem so long with you i
 see forever oh so clearly i might have been in love before
 but it never felt this strong our dreams are young and we
 both know they'll take us where we want to go hold me
 now touch me now i don't want to live without you noth-
 ing's gonna change my love you for you you oughta
 know by now how much i love you one thing you
 can be sure of i never ask for more than your love
 nothing's gonna change my love for you you
 oughta know by now how much i love
 you you'll only chang my whole
 life throug but nothing's
 gonna chang my
 love for you
 ♡
 一
 闪
 一闪
 亮晶晶
 留下岁
 月的痕迹
 我的世界的
 中心依然还是
 你一年一年又一年飞逝仅在一转眼唯一永远不改变
 是不停地改变我不像以前的自己你也有点不
 像你但在我的眼中你的笑依然的美丽
 日子只能往前走一个方向顺时
 钟不知道能爱多久所以
 要让你懂我依然爱你就
 是唯一的退路我依然珍惜
 时时刻刻的幸福你每个
 呼吸每个动作每个
 表情到最后一
 定会依然
 爱你你

Figure 4: L^AT_EX 可以生成很有趣的文字特效

2 最简单的文档示例

先看一个最简单的文档示例：

```

1 \documentclass{article}
2 % 引言区
3 \begin{document}
4 % 正文区
5 Hello \LaTeX{} !
6 \end{document}

```

编译的结果是：

Hello \LaTeX !

2.1 分析

2.1.1 文档类型

当 \LaTeX 处理源文件时，首先需要知道的是作者所要创建的文档类型。该信息可以通过命令 `\documentclass` 来提供给 \LaTeX 。

```

1 \documentclass[options]{class}

```

- 常用的文档类 (class)

Table 1: 文档类

类别	一般用途	特性
article	一般短文	无章，连续页式的安排，无奇偶数页的区分
report	较长论文	章会起新页，预设无奇偶数页的区分
book	排版书籍	章会于奇数页起新页，预设偶数页的区分
letter	信件	英文信件格式
slides	排版幻灯片	可以用其他宏包取代 (如 Beamer)
minimal	测试及写新类别	这是最简单的类别，只规定了内文的宽、高、正常字

- 文档类选项 (option)

文档类选项可以有多个，选项之间用逗号分隔。

Table 2: 文档类选项

选项	用途
10pt, 11pt, 12pt	设置文档所使用的字体的大小。如果没有声明任何选项，缺省将使用 10pt 字体。
a4paper, letterpaper, ...	定义纸张的大小，缺省的设置为 letterpaper。此外，还可以使用 a5paper, b5paper, executivepaper 和 legalpaper。
fleqn	设置该选项将使数学公式左对齐，而不是中间对齐。
leqno	设置该选项将使数学公式的编号防置于左侧。
titlepage, notitlepage	指定是否在文档标题 (document title) 后开始一新页。article 文档类缺省不开始新页，而 book 文档类则相反。

选项	用途
onecolumn, twocolumn	指定 L ^A T _E X 以单列 (one column) 或双列 (two column) 方式排版文档。
twoside, oneside	指定 L ^A T _E X 排版的文档为双面或单面格式。article 和 report 缺省使用单面格式, 而 book 则缺省使用双面格式。需要注意的是该选项仅作用于文档的式样。twoside 选项不会通知你的打印机让以得到双面的打印输出。
openright, openany	此选项决定新的章是仅仅在右边页 (奇数页) 还是在下一可用页开始。该选项对 article 文档类不起作用, 因为该类中并没有定义“章”(Chapter)。report 类中新的一章开始于下一可用页, 而 book 类中新的一章总是开始于右边页。

2.1.2 引言 (preamble)

在 `\documentclass` 语句和 `\begin{document}` 之间的区域称为 **引言区 (preamble)**, 这里可以通过 `\usepackage` 语句引入一些宏包, 或者定义一些全局的命令和格式。

3 几个小技巧

3.1 获取帮助

要获取对某个宏包的帮助，可以使用 `texdoc` 命令，该命令将为你打开该宏包的帮助文档。试试打开 Windows 下的 CMD 或者 Linux 下的终端，然后输入以下命令：

```
1 texdoc xetex
```

你可以把“xetex”换成其他你想要获取帮助的宏包。

3.2 编译当前一小段

只对选中的一小段进行编译，这样不仅能节省时间，尽快看到新增内容的编译结果，还有助于检查语法错误。

在 WinEdt 中编译一小段的方法很简单：

1. 选中要编译的 TEX 代码
2. 选择 Accessories → Compile Selected, (快捷键 `Ctrl` + `Shift` + `C`)
3. 会弹出 L^AT_EX 编译后的效果

3.3 清理辅助文件

L^AT_EX 文档在编译时会产生很多辅助文件，这些文件不是我们直接需要的 pdf 文档，所以可以清理掉。

两种方法：

1. 点击 WinEdt 的回收站图标，可以选择把 L^AT_EX 生成的辅助文件删除，方便快捷。
2. 将以下文本另存为 `clean.bat` 文件，放在源文件夹中，双击即可。清除的文件还包括子文件夹。

```
1 del *.aux /s
2 del *.bak /s
3 del *.log /s
4 del *.bbl /s
5 del *.dvi /s
6 del *.blg /s
7 del *.thm /s
8 del *.toc /s
9 del *.out /s
```

3.4 注释

3.4.1 单行注释 (%)

当 L^AT_EX 在处理源文件时，如果遇到一个百分号字符 %¹，那么 L^AT_EX 将忽略 % 后的该行文本，分行符以及下一行开始的空白字符。这样，我们就可以在源文件中写一些注释，而不会担心他们会出现在最后的排版结果中。

¹ % 也可以用来分割不允许有空格或分行的较长输入文本。

```

1 This is an % stupid
2 % Better: instructive <--
3 example: Supercal%
4         ifragilist%
5 icexpialidocious

```

编译结果:

This is an example: Supercalifragilisticexpialidocious

3.4.2 多行注释

如果需要较长的注释,可以在要注释的两段代码之间加上 `\iffalse ... \fi`, 例如:

```

1 这是另一个
2 \iffalse
3 非常笨拙的,
4 然而却很有效。
5 \fi
6 在你的文档中
7 使用注释的例子。

```

编译结果:

这是另一个 在你的文档中使用注释的例子。

3.5 英文断字的处理

必要时就会出现断字。如果断字算法不能确定正确的断字点,可以使用几种方法告诉 \TeX 如何弥补这个缺憾。

3.5.1 使用命令 `\hyphenation`

```
\hyphenation{word list}
```

使列于参量中的单词仅在注有“-”的点断字。命令的参量仅由正常字母构成的单词,或由激活文本中视为正常字母的符号组成。应用于(特殊)语言的已存好,当断字命令出现时,就为激活的语言储存断字可选点。这意味着如果你在文档导言中设置了断字命令,它将影响英文的断字。如果断字命令置于 `\begin{document}` 后面,而且你正使用类似 `babel` 的国际语言支持宏包,那么断字可选点在由 `babel` 激活的语言中就处于活动状态。

下面的例子允许对“`hyphenation`”和“`Hyphenation`”进行断字,却根本不允许“`FORTTRAN`”,“`Fortran`”和“`fortran`”进行断字。在参量中不允许出现特殊的字符和符号。

```
1 \hyphenation{FORTTRAN Hy-phen-a-tion}
```

命令 `\-` 在单词中插入一个自主的断字点。它也就成为这个单词中允许出现的唯一断字点。对于包含特殊字符(注音字符)的单词,这个命令是特别有用的,因为对于包含特殊字符的单词 \LaTeX 不自动断字。

```

1 I think this is: su\-\per\-\cal\-\%
2 i\-\frag\-\i\-\lis\-\tic\-\ex\-\pi\-\%
3 al\-\i\-\do\-\cious

```

编译结果是:

I think this is: supercalifragilisticexpialidocious

3.5.2 使用盒子

使用盒子也可以保证把几个单词排在同一行上。所谓“盒子”，就是把放在盒子里面的对象（小到一个字母，大到一个段落）当成一个整体来对待，这样，这个盒子就和一个字母一样具有了不可分割的原子性。

最简单的盒子命令是 `\mbox` 和 `\fbox`。前者把一组对象组合起来，后者在此基础上加了个边框。例如：

```
1 My phone number will change soon.
2 It will be \mbox{0116 291 2319}.
3
4 The parameter
5 \fbox{\emph{filename}} should
6 contain the name of the file.
```

编译结果是：

My phone number will change soon. It will be 0116 291 2319.

The parameter *filename* should contain the name of the file.

`\fbox` 还常常用来给插图加上边框，这个将在3.7.4讨论。

3.6 断行和分页

- 另起一段：在两段文字之间空一行空行。
- 另起一行，而不另起一段：

```
\\ or \newline
```

- 在强行段行后，还禁止分页：

```
\\*
```

- 另起一新页：

```
\newpage
```

3.7 插图技巧

3.7.1 插图方法

引入 **graphicx** 宏包后，使用 `\includegraphics{file}` 命令可以插入图片文件。在期刊写作中，插图常常需要放在 **figure** 浮动体中。之所以称为浮动是指最终排版效果图形的位置不一定与源文件中的位置对应，这也是刚使用 **L^AT_EX** 同学可能遇到的问题。如果要强制固定浮动图形的位置，请使用 **float** 宏包，它提供了 [H] 参数，但是除非特别需要，不建议使用 [H]，而是倾向于使用 [htbp]，给 **L^AT_EX** 更多选择。例如：

```
1 \begin{figure} [htbp]
2   \centering
3   \includegraphics[width=.6\textwidth]{tyzw.jpg}
4   \caption{藤原 ~ 佐为}
5   \label{fig:ty}
6 \end{figure}
```

编译结果见图5。



Figure 5: 藤原 佐为

对于插图比较多的文档，可以统一将插图放在一个 `figures` 目录，以方便管理。将下面这句话放在引言区：

```
1 \graphicspath{{figures/}}
```

3.7.2 多个图形

如果多个图形相互独立，并不共用一个图形计数器，那么用 `minipage` 或者 `parbox` 就可以。否则，请参看图 6，它包含两个小图，分别是图 6a 和图 6b。推荐使用 `subfig` 宏包，它提供 `\subfloat` 命令，不要再用 `\subfigure` 和 `\subtable`。

```
1 \begin{figure}[htb]
2   \centering%
3   \subfloat[第一个小图形]{%
4     \label{fig:subfig1}
5     \includegraphics[height=3cm]{jtg.jpg}}\hspace{4em}%
6   \subfloat[第二个小图形。如果标题很长的话，它会自动换行，这个 caption 就是这样的例子]{%
7     \label{fig:subfig2}
8     \includegraphics[height=3cm]{tsl.jpg}}
9   \caption{包含子图形的大图形}
10  \label{fig:big1}
11 \end{figure}
```

编译结果见图6。

下面这个例子显示并排 3×2 的图片：



Figure 6: 包含子图形的大图形

```

1 \begin{figure} [htb]
2 \centering
3 \subfloat[] {\includegraphics[width=.27\textwidth]{qh2.jpg}} \quad
4 \subfloat[] {\includegraphics[width=.27\textwidth]{qh2.jpg}} \quad
5 \subfloat[] {\includegraphics[width=.27\textwidth]{qh2.jpg}} \quad
6 \subfloat[] {\includegraphics[width=.27\textwidth]{qh2.jpg}} \quad
7 \subfloat[] {\includegraphics[width=.27\textwidth]{qh2.jpg}} \quad
8 \subfloat[] {\includegraphics[width=.27\textwidth]{qh2.jpg}}
9 \caption{并排图片}
10 \label{fig:subfig:3x2}
11 \end{figure}

```

编译结果见图7:

3.7.3 旋转和缩放图案

通过使用 `\includegraphics` 的 `angle` 选项可以旋转图片。如果要缩放图片，可以通过 `scalebox` 环境来实现，例如：

```

1 \begin{figure}
2 \centering
3 \subfloat[原始图像] {\includegraphics{tyzw2}}
4 \subfloat[旋 转      $30^\circ$] {\begin{minipage} [b] {.3\linewidth} \includegraphics[origin=c,angle=30]{tyzw2} \end{minipage}} \quad
5 \subfloat[水 平 翻 转] {\begin{minipage} [b] {.3\linewidth} \scalebox{-1}[1] {\includegraphics{tyzw2}} \end{minipage}}
6 \subfloat[垂 直 翻 转] {\begin{minipage} [b] {.3\linewidth} \scalebox{1}[-1] {\includegraphics{tyzw2}} \end{minipage}}
7 \subfloat[缩 放      (0.5,1.2)] {\begin{minipage} [b] {.3\linewidth} \scalebox{-0.5}[1.2] {\includegraphics{tyzw2}} \end{minipage}}
8 \end{figure}
9 \caption{旋转和缩放图案示例}
10 \label{fig:tyzw2}

```

编译结果见图8。



Figure 7: 并排图片

3.7.4 给图像加边框

使用 `\fbox` 可以给图像加边框，可以设置 `fboxsep` 和 `\fboxrule` 来分别调整边框边距和粗细。例如：

```

1 \begin{figure}
2 \centering
3 \setlength{\fboxsep}{0pt}
4 \subfloat[加边框图像]{\fbox{\includegraphics{tyzw2}}}
5 \setlength{\fboxrule}{10pt}
6 \subfloat[10pt 粗的边框效果]{\fbox{\includegraphics{tyzw2}}}
7 \caption{加边框图像}
8 \label{fig:boxedfigures}
9 \end{figure}

```

编译结果见图9。

通过引用 **fancybox** 宏包，还可以使用更多的盒子来实现更丰富的边框效果，可选的盒子包括：

小圆角盒子 大圆角盒子 双线盒子 阴影盒子

例如：



(a) 原始图像



(b) 旋转 30°



(c) 水平翻转



(e) 缩放 (0.5,1.2)

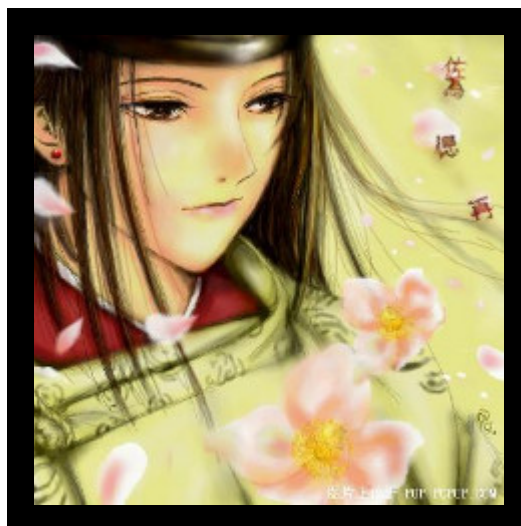


(d) 垂直翻转

Figure 8: 旋转和缩放图案示例



(a) 内边距为 0 的加框图像



(b) 10pt 粗的边框效果

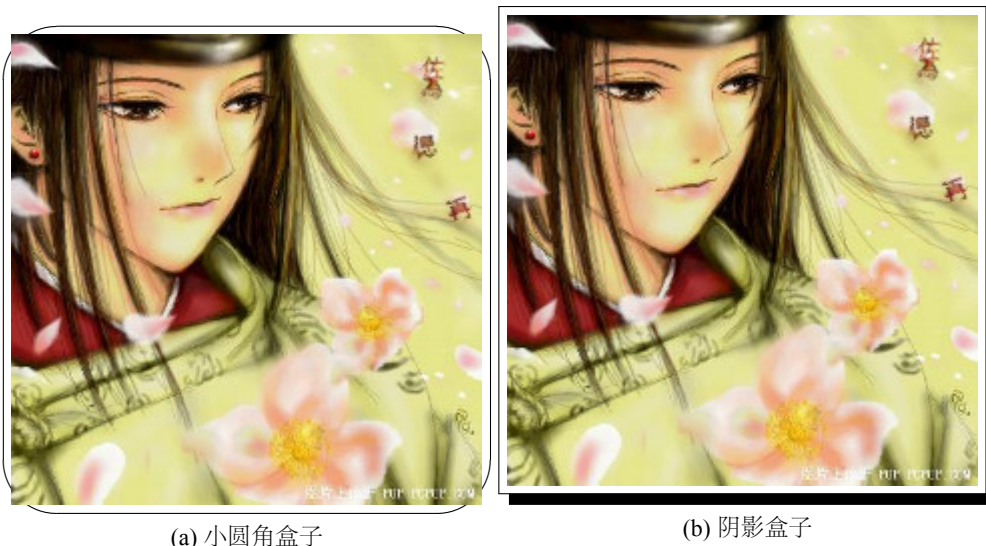
Figure 9: 加边框图像

```

1 \begin{figure}
2 \centering
3 \subfloat[小圆角盒子]{\ovalbox{\includegraphics{tyzw2}}}
4 \setlength{\fboxrule}{.4pt}
5 \subfloat[阴影盒子]{\shadowbox{\includegraphics{tyzw2}}}
6 \caption{更多边框效果}
7 \label{fig:fancyboxedfigures}
8 \end{figure}

```

编译结果见图10。



(a) 小圆角盒子

(b) 阴影盒子

Figure 10: 更多边框效果

4 初学者要注意的地方

4.1 特殊字符

当心!!! 下面的这些字符是 \LaTeX 的保留字符, 它们或在 \LaTeX 中有特定的用处, 或不一定包含在所有的字库中。如果你直接在文本中使用它们, 通常在排版结果中将不会得到这些字符, 而且还会导致 \LaTeX 做一些你不希望发生的事情。

\$ % ^ & _ { } ~ \

大部分特殊字符在前面加上反斜线 “\”, 就可以在文本中得到它们。但对于反斜线 “\”, 要得到它, 可以使用 $\text{\$}\backslash\text{\$}$ 。

4.2 标点符号

4.2.1 引号

- 英文引号

这个是初学者最容易犯的错误之一!!! 在 \LaTeX 中, 英文的引号 **不能** 通过直接输入” 来得到。在 \LaTeX 中, 左边的引号是 grave accent, 是键盘左上方 **[Esc]** 或 **[F1]** 下方有波形号的那个键; 右边的是 apostrophe, 也就是主键盘区的 **[Enter]** 键隔壁的那个键。双引号的情形是键入两次的左单引号以及两次的右单引号, 而不是用” 这个一次完成两个点的 ditto marks。示例:

```
1 Please press an ‘Esc’ key.\\
2 Please press an ’ Esc’ key. 这是错误示范!\\
3 ‘This sentence.’ \\
4 "This sentence." 这是错误示范!
```

编译结果是:

```
Please press an ‘Esc’ key.
Please press an ’ Esc’ key. 这是错误示范!
```

“‘This sentence.’”
 ”This sentence.” 这是错误示范!

- 中文引号
 英文引号的问题在中文排版中则不存在，因为中文使用全角标点。所以在 \LaTeX 中输入双引号时，只需要依次输入 ‘ ‘ 和 ‘ ’ ’ 即可。

4.2.2 省略号

- 英文省略号
 英文的省略号是三点，如果碰到句号的话，就是四点。但是英文的这个三点不能直接打三个句号就完事，否则这样打出来的省略号会显得太过密集。可以使用 \ldots 或 \dots 指令，例如：

```
1 I' m not a good man ..., but a good husband .... 错误示范!\\
2 I' m not a good \ldots\ man \ldots, but a good husband \ldots.\\
3 I' m not a good \dots\ man \dots, but a good husband \dots.
```

编译结果是：

I' m not a good ... man ..., but a good husband 错误示范!
 I' m not a good ... man ..., but a good husband
 I' m not a good ... man ..., but a good husband

- 中文省略号
 中文省略号是六点，同样由于是全角字符，所以直接输入 ‘……’ 即可。

4.2.3 破折号

- 英文破折号
 \LaTeX 中右四种短划标点符号。连续用不同数目的短划，可以得到其中的三种。第四个实际不是标点符号，它是数学中的减号：

```
1 daughter-in-law, X-rated\\
2 pages 13-67\\
3 yes-or no? \\
4 $0$, $1$ and $-1$
```

编译结果是：

daughter-in-law, X-rated
 pages 13-67
 yes—or no?
 0, 1 and -1

- 中文破折号
 中文的破折号比较特殊，可以通过画一条长为两个字宽（ 2em ）的线来实现。为了方便处理，可以专门定义一个破折号的命令，将这一句话

```
\newcommand{\pozhehao}{\kern0.3ex\rule[0.8ex]{2em}{0.1ex}\kern0.3ex}
```

放在引言区 2.1.2 里，然后在文中需要用破折号的地方直接使用 \pozhehao 来插入一个中文破折号。例如：

```
1  艰苦奋斗、严谨治学、求实创新、为人师表 \hfill \pozhehao{} 华南师范大学校训
```

编译结果是：

艰苦奋斗、严谨治学、求实创新、为人师表 ——华南师范大学校训

4.2.4 波浪号 (~)

比较常用的波浪号 \sim 通常使用 \sim 产生。注意使用 \sim 产生的是比较小的波浪号 \sim 。

4.2.5 度的符号 (°)

可以通过下面的语句得到这个符号，本质上就是绘制一个上标形式的圆圈 (\circ):

```
1  Its  $-30^\circ$ , I will soon start to
2  super-conduct.
3
```

编译结果是：

Its -30° C, I will soon start to super-conduct.

4.2.6 数学符号

如果从其它地方拷贝文字，**千万不要忘记把数学符号改成 L^AT_EX 下的形式**，否则编译后这些符号可能变得不可见。

5 中文处理

推荐使用 **xeCJK** 来处理中文，首先需要在引用区加入下面的命令：

```

1 \RequirePackage[CJKnumber,CJKtextspaces,CJKmathspaces,no-math,BoldFont,Slant-
  Font]{xeCJK}
2 \RequirePackage[indentfirst]
3 \RequirePackage[numbers,sort&compress,square]{natbib}
4 \defaultfontfeatures{Mapping=tex-text}
5 \xeCJKsetcharclass{"0"}{"2E7F"}{0}
6 \xeCJKsetcharclass{"2E80"}{"FFFF"}{1}

```

然后需要配置中文字体，推荐使用 **Adobe** 的字体方案：

```

1 \setmainfont{Times New Roman}
2 \setsansfont{Arial}
3 \setmonofont{Courier New}
4 \setCJKmainfont[BoldFont={Adobe Heiti Std},ItalicFont={Adobe Kaiti Std}]{Ado-
  be Song Std}
5 \setCJKsansfont{Adobe Heiti Std}
6 \setCJKmonofont{Adobe Kaiti Std}
7 \setCJKfamilyfont{song}{Adobe Song Std}
8 \setCJKfamilyfont{hei}{Adobe Heiti Std}
9 \setCJKfamilyfont{fs}{Adobe Fangsong Std}
10 \setCJKfamilyfont{kai}{Adobe Kaiti Std}
11 \setCJKfamilyfont{li}{Adobe Kaiti Std}
12 \setCJKfamilyfont{you}{Adobe Kaiti Std}
13
14 \newcommand{\song}{\CJKfamily{song}}      % 宋体
15 \newcommand{\fs}{\CJKfamily{fs}}         % 仿宋体
16 \newcommand{\kai}{\CJKfamily{kai}}      % 楷体
17 \newcommand{\hei}{\CJKfamily{hei}}      % 黑体
18 \newcommand{\li}{\CJKfamily{li}}        % 隶书
19 \newcommand{\you}{\CJKfamily{you}}      % 幼圆
20 \def\songti{\song}
21 \def\fangsong{\fs}
22 \def\kaishu{\kai}
23 \def\heiti{\hei}
24 \def\lishu{\li}
25 \def\youyuan{\you}

```

之后可以设置中文的字体大小方案，这里使用了清华大学薛瑞尼的设置：


```

1 \newlength\thu@linespace
2 \newcommand{\thu@choosefont}[2]{%
3   \setlength{\thu@linespace}{#2*\real{#1}}%
4   \fontsize{#2}{\thu@linespace}\selectfont}
5 \def\thu@define@fontsize#1#2{%
6   \expandafter\newcommand\csname #1\endcsname[1][\baselinestretch]{%
7     \thu@choosefont{##1}{#2}}
8 \thu@define@fontsize{chuhao}{42bp}
9 \thu@define@fontsize{xiaochu}{36bp}
10 \thu@define@fontsize{yihao}{26bp}
11 \thu@define@fontsize{xiaoyi}{24bp}
12 \thu@define@fontsize{erhao}{22bp}
13 \thu@define@fontsize{xiaoer}{18bp}
14 \thu@define@fontsize{sanhao}{16bp}
15 \thu@define@fontsize{xiaosan}{15bp}
16 \thu@define@fontsize{sihao}{14bp}
17 \thu@define@fontsize{banxiaosi}{13bp}
18 \thu@define@fontsize{xiaosi}{12bp}
19 \thu@define@fontsize{dawu}{11bp}
20 \thu@define@fontsize{wuhao}{10.5bp}
21 \thu@define@fontsize{xiaowu}{9bp}
22 \thu@define@fontsize{liuhao}{7.5bp}
23 \thu@define@fontsize{xiaoliu}{6.5bp}
24 \thu@define@fontsize{qihao}{5.5bp}
25 \thu@define@fontsize{bahao}{5bp}

```

除此之外，还可以对将图、表等页面元素中文化：

```

1 \renewcommand\figurename{图}
2 \renewcommand\tablename{表}

```

完成配置之后，就可以灵活使用各种字体：

1 华南师范大学历史上的名字包括：

```

2
3 \begin{itemize}
4 \item[楷体] {\kai 广州市立师范学校}
5 \item[黑体] {\hei 勳勤大学师范学院}
6 \item[隶书] {\li 勳勤大学教育学院}
7 \item[宋体] {\song 广东省立教育学院}
8 \item[仿宋] {\fs 广东省立文理学院}
9 \item[粗体] {\bfseries 广东省文理学院}
10 \item[斜体] {\itshape 华南师范学院}
11 \item[粗斜体] {\bfseries\itshape 广东师范学院}
12 \item[字体颜色] {\textcolor{red}{华南师范大学}}
13 \end{itemize}

```

华南师范大学历史上的名字包括：

楷体 广州市立师范学校

黑体 勳勤大学师范学院

隶书 勳勤大学教育学院

宋体 广东省立教育学院

仿宋 广东省立文理学院

粗体 广东省文理学院

斜体 华南师范学院

粗斜体 广东师范学院

字体颜色 华南师范大学

上面这段内容使用了 `itemize` 列表环境， \LaTeX 默认的列表环境会在条目之间插入过多的行距，若用户需要紧凑的行距，可以使用 `compactitem` 环境。

下面测试英文字体：

```

1 Remember the \textsf{more} \textbf{font} {\bfseries\tiny you \sffamily use,
2 \Large the \scshape more \itshape beautiful \slshape your \footnotesize
3 document becomes.}
4
5 \begin{itemize}
6 \item[英文黑体] Typeset text in \textbf{bold} series
7 \item[英文斜体] Typeset text in \textit{italic} shape
8 \item[Roman 字体] Typeset text in roman family
9 \item[Sans Serif 字体] Typeset text in \textsf{sans serif} family
10 \item[typewriter 字体] Typeset text in \texttt{typewriter} family
11 \end{itemize}

```

Remember the more font you use, **the more beautiful your document becomes.**

英文黑体 Typeset text in **bold** series

英文斜体 Typeset text in *italic* shape

Roman 字体 Typeset text in roman family

Sans Serif 字体 Typeset text in sans serif family

typewriter 字体 Typeset text in typewriter family

下面测试字号：

```

1 \begin{itemize}
2 \item[初号] {\chuhao 华南师范大学}
3 \item[小初] {\xiaochu 华南师范大学}
4 \item[一号] {\yihao 华南师范大学}
5 \item[小一] {\xiaoyi 华南师范大学}
6 \item[二号] {\erhao 华南师范大学}
7 \item[小二] {\xiaoer 华南师范大学}
8 \item[三号] {\sanhao 华南师范大学}
9 \item[小三] {\xiaosan 华南师范大学}
10 \item[四号] {\sihao 华南师范大学}
11 \item[小四] {\xiaosi 华南师范大学}
12 \item[五号] {\wuhao 华南师范大学}
13 \item[小五] {\xiaowu 华南师范大学}
14 \end{itemize}

```

初号 华南师范大学

小初 华南师范大学

一号 华南师范大学

小一 华南师范大学

二号 华南师范大学

小二 华南师范大学

三号 华南师范大学

小三 华南师范大学

四号 华南师范大学

小四 华南师范大学

五号 华南师范大学

小五 华南师范大学

6 参考文献

在 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 下管理参考文献将极其方便，建议使用 [Jabref](#) 生成条目，用 `\cite` 或者 `upcite` (其中 `upcite` 是上标索引) 索引即可。

6.1 JabRef 简介

JabRef 是一个用 Java 开发的文献库管理软件，用 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 写论文的时候与它配合是妙不可言的。官方主页在：<http://jabref.sourceforge.net>。运行它需要先安装了 [Java 运行环境](#)。

启动 JabRef 后，第一步是给你的论文创建一个文献库。输入 `[Ctrl] + N` 组合键，或者点击 **【File】** → **【New Database】**，新建一个文献库。建议保存在论文同一个目录下，名字随意，例如 `refs.bib`。

6.2 利用 Google Scholar 获取 BibTeX

建议养成一边阅读文献一边维护文献库的好习惯。当看到有用的文献材料，就将它的 BibTeX 保存下来，导入 JabRef 生成条目。这样，当你需要引用文献的时候，就可以很方便地通过 `Bibtexkey` 来引用该文献对应的条目，而不需要自己编写 `BibItem`。

[Google Scholar](#) 是一个强大的文献检索工具，可以很方便的获得检索结果的 BibTeX。但需要对 Google Scholar 进行设置。进入 Google Scholar 后，首先点击页面右上角的“Settings”图标进入设置，在页面最下方的“Bibliography manager”中，勾选“Show links to import citations into BibTeX”选项。如图 11 所示。

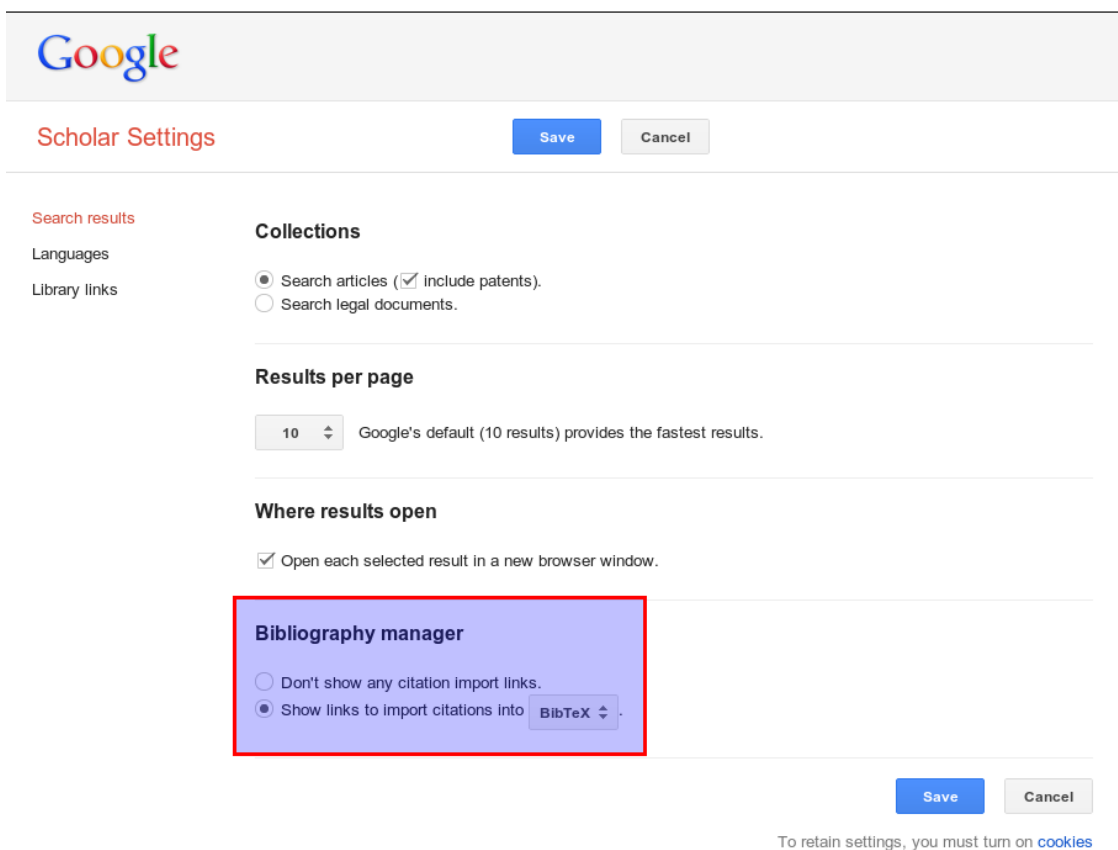


Figure 11: 设置 Google Scholar

完成后点击“保存”按钮回到 Google scholar 主页，此时会发现文献检索结果中会多出一个【import into BibTeX】选项，如图 12所示。

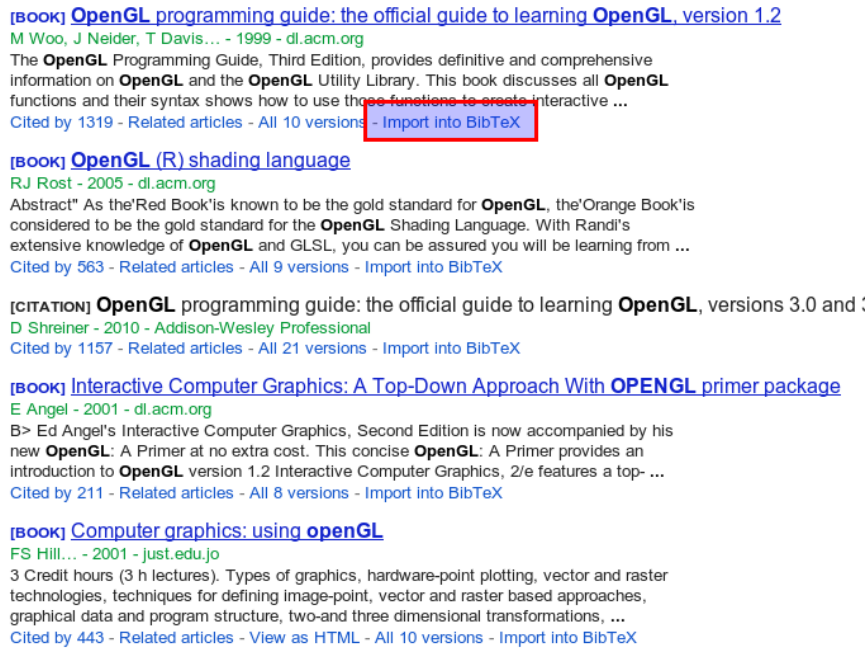


Figure 12: import into BibTeX

点击进去可以看到该文献的 BibTeX 内容，将该内容保存成一个文件。然后在 JabRef 中按 **Ctrl** + 组合键导入该文件。也可以直接复制这段内容，然后在 JabRef 中按 **Ctrl** + **Shift** + **N** 组合键打开一个新建条目窗口，然后把这段内容粘贴进去。如图 13所示。

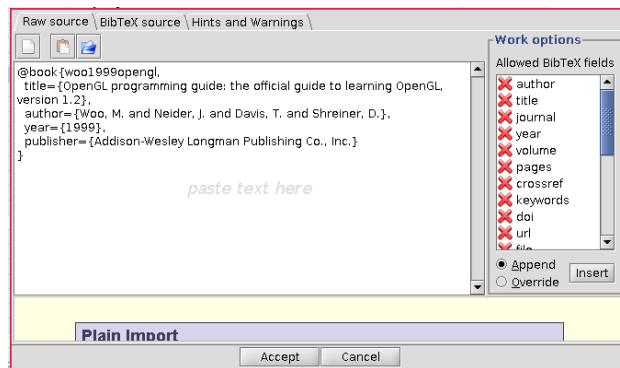


Figure 13: 新建条目并粘贴 BibTeX 信息

导入成功后，新建的文献条目将会以浅绿色高亮显示。如果有需要，还可以双击该条目进行编辑，甚至还可以为它添加若干个文件链接，将该文献条目与对应的论文的路径连接起来，从而形成一个名副其实的文献管理库。

6.3 在 \LaTeX 中引用文献

在 \LaTeX 中引用文献很简单，只需要完成两步设置：

- 1 `\bibliographystyle{xxx}`
- 2 `\bibliography{yyy}`

其中，`\bibliographystyle` 设置引用样式，常用的“xxx”是引用样式的名字，该值可以是 `plain`、`abbrv`、`siam`、`alpha` 或者表 3 列举的各种引用样式。

Table 3: 各种引用格式一览表

1: ieeetr	2: unsrt	3: IEEE	4: ama	5: cj
6: nar	7: nature	8: phjcp	9: is-unsrt	10: plain
11: abbrv	12: acm	13: siam	14: jbact	15: amsplain
16: finplain	17: IEEEannot	18: is-abbrv	19: is-plain	20: annotation
21: plainyr	22: decsci	23: jtbnew	24: neuron	25: cell
26: jas99	27: abbrvnat	28: ametsoc	29: apalike	30: jqt1999
31: plainnat	32: jtb	33: humanbio	34: these	35: chicagoa
36: development	37: unsrtnat	38: amsalpha	39: alpha	40: annotate
41: is-alpha	42: wmaainf	43: alphanum	44: apasoft	

要查看这些样式的效果，可以查看 [bibstyles.pdf](#) 文件。大部分的样式都已经集成在大部分的 L^AT_EX 系统中了，其他的可以从 <http://www.tug.org/tex-archive/biblio/bibtex/contrib/> 下载。所有的引用样式都是后缀名为 `.bst` 的文件。例如，要使用 `abbrvnat` 样式，你必须保证在你的 L^AT_EX 系统安装目录下有 `abbrvnat.bst` 文件，如果没有的话，你可以下载一个，然后放在你的工作目录下，或者其他 L^AT_EX 能够找到的地方。

设置好样式后，第二步是使用 `\bibliography` 设置文献库路径，将“yyy”改为你用 JabRef 创建和维护的那个文献库的地址。

接下来要做的就是需要在需要引用文献的地方使用 `\cite{bibtexkey}` 或者 `\upcite{bibtexkey}` 将文献引用进来，其中，`upcite` 是上标引用，你需要事先定义这个样式：

```
1 \newcommand{\upcite}[1]{\textsuperscript{\cite{#1}}}
```

`bibtexkey` 则为你要引用的文献的 `Bibtexkey`，在 JabRef 中可以直接通过按下 **Ctrl** + **Shift** + **k** 组合键获得 `Bibtexkey`，或者直接通过按下 **Ctrl** + **k** 组合键获得形式为 `\cite{bibtexkey}` 的引用命令，然后粘贴进 L^AT_EX 文档中。

对于中文文献的处理，推荐使用北邮的 `bstutf8.bst` 样式，这里有一个下载地址：<http://zzuthesis.googlecode.com/svn-history/r11/trunk/bstutf8.bst>。另外，如果出现乱码错误，则需要设置好文献库的编码。在 JabRef 中点击 **File** → **Data Properties** 菜单项，打开数据属性设置窗口，将 **Database encoding** 选项设为 UTF8 即可。如图 14 所示。

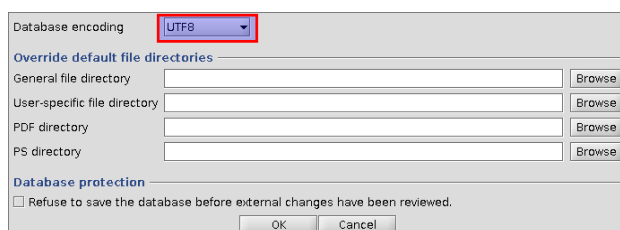


Figure 14: 设置文献库编码

注意在编译文档时，记得 BibTeX 一下，生成 BibItem。

Index

TeX 系统

CTeX, 8
fpTeX, 8
MikTeX, 8
teTeX, 8
TeXLive, 8

命令

\\, 16
*, 16
\bibliography, 30
\bibliographystyle, 30
\caption, 16
\cite, 29
\dots, 23
\doublebox, 19
\fbox, 16
\fi, 15
\graphicspath, 17
\hyphenation, 15
\iffalse, 15
\includegraphics, 16
\label, 16
\ldots, 23
\mbox, 16
\newpage, 16
\Ovalbox, 19
\ovalbox, 19
\shadowbox, 19
\subfigure, 17
\subfloat, 17
\subtable, 17

宏包

article, 12
Asymptote, 9
book, 12
fancybox, 19
float, 16
gnuplot, 9
graphicx, 16
letter, 12
MetaPost, 9
minimal, 12
report, 12
scalebox, 18
slides, 12
subfig, 17

subfigure, 17
TikZ, 9
xeCJK, 25

环境

figure, 16
minipage, 17
parbox, 17

选项

\fboxrule, 19
\fboxsep, 19
a4paper, 12
a5paper, 12
angle, 18
b5paper, 12
executivepaper, 12
fleqn, 12
H, 16
htbp, 16
legalpaper, 12
leqno, 12
letterpaper, 12
notitlepage, 12
oneside, 12
openany, 12
openright, 12
titlepage, 12
twoside, 12