

附录B 参数文献数据库的处理

在科技出版物的创作中，参考文献的组织是一个不可缺少的标准过程。我们已经在 4.3.6 节和 8.3.2 节中讲解了如何利用 `thebibliography` 环境来排版参考文献，以及如何在正文中引用其中的条目。有时候作者会发现在绝大多数文章中他经常引用的是同样作品。与此类似，在同一个领域工作的研究人员参考的论文也大致相同。这也就是说在不同文档的 `thebibliography` 环境中经常有大量重复的条目，在一个研究所中的同事与同事之间处理的文档里也有类似的情形。

如果我们能把参考文献条目全部存贮在一个数据库文件里，使得所有的文档都可以引用其中的一组文献，这就会节省大量的时间。借助于 \LaTeX 宏包中的 `BibTeX` 程序我们就可以建立起这样的一个数据库。把各种出版物的信息存贮在一个或多个后缀为 `.bib` 的文件中。每篇文献有一个 关键词 来与其它文献区分开，在正文中可以通过它来引用该文献。这样的文件就称为 参考文献数据库。

§B.1 BibTeX 程序

`BibTeX` 是 \LaTeX 的一个辅助程序，它通过搜索一个或多个数据库，自动为 \LaTeX 文档构造参考文献。要做到这一点， \LaTeX 文件中必须在参考文献所位于的地方调用命令

```
\bibliography{数据库一, 数据库二, ... }
```

其中的参数值 数据库一, 数据库二, ... 就是要搜索的参考文献数据库的基本名，中间用逗号分开。并不需要显式地给出 `.bib` 后缀。

在 \LaTeX 正文任何地方都可以通过下面命令引用数据库中的一篇文献：

```
\cite{关键词}
```

这与 8.3.2 节中的说明一样。其中 关键词 就是文献的标志，这当然要求用户必须预先知道这一信息。在第一次运行 \LaTeX 后，就必须执行 `BibTeX` 程序。至于如何调用这个程序，那就要看所处的计算机操作系统了，但通常的方法就是操作命令 `bibtex` 后接 \LaTeX 文件的基本名。假设这个基本名是 `comets`，那么

```
bibtex comets
```

就会生成一个名为 `comets.bbl` 的新文件，其中就包含从数据库中提取的相应于 `\cite` 引用的参考文献信息，并包装到一个 `thebibliography` 环境中，下次运行 \LaTeX 时就会把它输入到文档中。

有时候需要包含进一个在正文中没有引用的文献。这可以用如下命令来加入该文献：

`\nocite{关键词}`

可以把这条命令放在主文档内的任意地方。它并不生成任何文本，只是告诉 BibTeX 也要把这一项放到参考文献中。而命令 `\nocite{*}` 会把数据库中所有项都包含到文档中，这对于要生成所有条目及其关键词是非常有用的。（** 对 BibTeX 0.98 版及以前版本这条命令是没有用的。 **）

在执行 BibTeX 后，就会生成 .bbl 文件，这就需要至少执行两次 LaTeX 以建立起参考文献和正文中正确的引用记号。参考文献就显示在 `\bibliography` 命令被调用的地方；这条命令实际上是把 .bbl 文件包含到正文中。

可以用下面的声明来选择参考文献的样式：

`\bibliographystyle{样式}`

可以在导言的任何地方调用这条命令。样式参数可以取如下的几个值：

- plain** 参考文献中的条目按字母顺序排列；每项条目都有一个活动编号，并放在方括号内，这就是在正文中引用该文献（即 `\cite` 命令调用的地方）时的标记。
- unsorted** 条目按第一次出现在 `\cite` 和 `\nocite` 命令中的顺序排列。在第一个 `\cite` 中的条目编号为 1，下一个具有不同关键词的 `\cite` 条目编号为 2，依此类推。其它标记和列表方式与 **plain** 样式相同。
- alpha** 参考文献的顺序与 **plain** 样式相同，但是记号用的是作者姓名的缩写加上出版年代。对 Smith(1987) 的引用将会是 [Smi87]。
- abbrev** 顺序和标记与 **plain** 相同，但是参考文献中要对作者的名字，月份，以及杂志名称进行缩写，以缩短长度。

在用户使用的计算机上可能还有其它可用的样式。特别地，正如 B.3.1 节所描述的 作者-年代 索引，我们可以从标准 LaTeX 宏包中衍生出大量的不同样式。参考文献样式是包含在后缀为 .bst 的文件中。

下一节就讲解供 BibTeX 程序 0.99 版本使用的参考文献数据库。在这个版本中有很多 0.98 版本所没有的功能，我们用双星号 ** 来标明这些新功能。那些在旧版本中可以使用的数据库文件 (.bib 文件) 在 0.99 版本中的功能仍然一样。然而，适用于一个版本的 .bst 参考文献样式文件在另一个版本中可能就行不通了。要想知道你所用的到底是哪个版本，只要看一下 BibTeX 运行时屏幕输出或草案文件 .bblg 的第一行。

§B.2 创建参考文献数据库

创建参考文献数据库看起来好象要比向 thebibliography 环境中输入一串引用文献的工作量大很多。但是这样做的最大优势就是一旦把某条目放到参考文献数据库中，也就一劳永逸了，以后所有的文章都可以引用它。即使以后需要另一种不同的参考文献样式，所有已在数据库中的条目也是可以用

的，这就比用另外的格式重写 thebibliography 环境要方便得多。实际上，据我们的经验，即使正在处理的只有一篇文章，向数据库中加入一项条目，也要比向文献列表中加入一项简单，因为后者需要准确而灵巧的定位，而且要考虑标点符号和作者名字的位置。如果有了在 B.2.6 节中演示的一般性模板，数据条目的处理就会变得非常简单，而且速度很快。

在参考文献中的条目具有如下的形式

```
@BOOK{knuth:86a,
  AUTHOR = "Donald E. Knuth",
  TITLE = {The \TeX{}book},
  EDITION = "third",
  PUBLISHER = "Addison--Wesley",
  ADDRESS = {Reading, Massachusetts},
  YEAR = 1986 }
```

第一个单词，其前缀 @，确定了条目类型，我们将在下节详细解释。条目类型后接放在大括号 { } 中的条目信息。其中第一项就是关键词，这也就是在 \cite 命令中引用的名称。在上面的例子中，其为 knuth:86a。关键词可以是字母、数字以及符号（逗号除外）的任意组合。然后就是真正的索引信息，它们分放在各个域中，相互之间用逗号分开。每个域由域名、一个等号 =（其两边可以有空格）以及域文本组成。上面给出的域名有 AUTHOR, TITLE, PUBLISHER, ADDRESS 和 YEAR。域文本必须放在大括号或者双引号内。然而，如果这部分文本只是由数字组成，比如上面的 YEAR 后的内容，那么可以不要括号或引号。

在输入条目时，有些域是不可少的，还有一些域则是可以省略的，而其它的则要被 BibTeX 忽略的。在下面与条目类型一起列出了各种域。如果缺少了某个不可少域，BibTeX 运行时就会给出一条错误消息。如果使用了可省域，那么它们在参考文献中就会包含所提供的信息，但是并不必要一定给出这些域的内容。而要被忽略的域可以在数据库中包含不输出的额外信息（例如注释或者论文摘要），这也是非常有用的。要被忽略的域也可以提供给其它的数据库程序使用。

在参考文献数据库中条目的一般语法为

```
@ 条目类型 { 关键词,
  域名 = { 域文本 }
  ....
  域名 = { 域文本 } }
```

在条目类型以及域名的名称中既可以用大写字母，也可以用小写字母，甚至是两者的混合。因此 @BOOK, @book, @book 都表示同样的类型。

在整个条目最外面的括号对既可以是如上所示的大括号 { }，也可以是

小括号 ()。对于后者, 一般的语法为

@条目类型(关键词,)

域文本 却只能放在大括号 {...} 或者双引号 "..." 内, 如上例所示。

§B.2.1 条目类型

下面按字母顺序列出了标准的条目类型, 并给出类型功能的简短描述, 以及其可包含的不可少域和可省略域名称。我们将在下一节解释域的意义。

@article 条目为期刊或杂志上的一篇文章。

不可少域 author, title, journal, year.

可省略域 volume, number, pages, month, note.

@book 条目为有确定出版社的书籍。

不可少域 author 或 editor, title, publisher, year.

可省略域 volume 或 number, series, address, edition, month, note.

@booklet 条目为印制的有封皮的作品, 但没有出版社或赞助机构的名称。

不可少域 title.

可省略域 author, howpublished, address, month, year, note.

@conference 与下面的 @inproceedings 相同。

@inbook 条目为一本书的一部分 (章, 节或某些页)。

不可少域 author 或 editor, title, chapter 和 / 或 pages, publisher, year.

可省略域 volume 或 number, series, type, address, edition, month, note.

@incollection 条目为一本书中有自己题目的一部分。

不可少域 author, title, booktitle, publisher, year.

可省略域 editor, volume 或 number, series, type, chapter, pages, address, edition, month, note.

@inproceedings 条目为会议论文集集中的一篇文章。

不可少域 author, title, booktitle, year.

可省略域 editor, volume 或 number, series, pages, address, month, organization, publisher, note.

@manual 条目为科技文档。

不可少域 title.

可省略域 author, organization, address, edition, month, year, note.

@mastersthesis 条目为硕士论文。

不可少域 author, title, school, year.

可省略域 type, address, month, note.

@misc 条目为不属于其它任何类型的作品。

不可少域 没有。

可省略域 `author`, `title`, `howpublished`, `month`, `year`, `note`.

`@phdthesis` 条目为博士论文。

不可少域 `author`, `title`, `school`, `year`.

可省略域 `type`, `address`, `month`, `note`.

`@proceedings` 条目为会议论文集。

不可少域 `title`, `year`.

可省略域 `editor`, `volume` 或 `number`, `series`, `address`, `month`,
`organization`, `publisher`, `note`.

`@techreport` 条目为学校或其它研究机构印制的报告。

不可少域 `author`, `title`, `institution`, `year`.

可省略域 `type`, `number`, `address`, `month`, `note`.

`@unpublished` 条目为有作者和标题的还未出版的作品。

不可少域 `author`, `title`, `note`.

可省略域 `month`, `year`.

在每项条目中还可以有可省略域 `key` 和 `crossref`。前者提供当没有作者信息时字母排序用的附加信息。作者信息通常就放在 `author` 域中，但也有可能放在 `editor` 域或者 `organization` 域中。这个 `key` 域与 `\cite` 命令中所用的识别条目的关键词没有任何关系。而 `crossref` 域给出另一个条目的关键词，使得两者共享某些域的信息，见 B.2.3 节。

§B.2.2 域

在一项参考文献条目中可能使用的域及其含义见下面的列表。而域的形式总是为

域名 = { 域文本 } 或者

域名 = " 域文本 "

`address`

出版社或者研究所的地址。对于大的出版社，只要给出所在的城市就可以了。而对于小的出版社，则建议给出详细的地址。

`annote` 在非标准参考文献样式中可以使用评注，这样可以得到有注释的参考文献。在标准 BibTeX 样式中这个域是被忽略的。

`author` 给出作者的姓名，见 B.2.4 节。

`booktitle`

当只引用了一本书的某一部分时，应给出书的题目。参考 B.2.4 一节中有关大写方面的特殊考虑。

`chapter`

章节的编号。

crossref

在数据库中另一项的关键词，这样两者可以共享一些相同的域文本。
见 B.2.3 节。

edition

书籍的版本，通常采用完整的首字母大写的单词表示，如 ‘Second’。
必要时标准样式可以把它改成小写。

editor 在 B.2.4 节描述的形式中的编辑姓名。如果还有 **author** 域，那么用这个域给出引用书籍或作品集的编辑。

howpublished

说明出版方法的非同寻常之处。应该首字母大写，如： ‘Privately published’。

institution

科技报告的举办机构。

journal

期刊或杂志的名称。对于那些相当常见的刊物名称，有相应的缩写（见 B.2.5 节）。

key 当没有给出作者信息时，用此信息进行字母排序。这与 `\cite` 命令中区分条目的 **关键词** 没有任何关联。

month 作品出版时的月份，或者指还未出版的作品写完时的月份。这里可以用月份（英文）名称的前三个字母进行缩写。

note 应加上的其它信息。首字母应该大写。

number 期刊，杂志，技术报告，或者一系列作品中的一部的编号。期刊通常用卷和期来标识；技术报告则由研究机构赋予一个编号；系列书籍中的某部有时也有一个编号。

organization

会议或报告的主办机构。

pages 页码或者起始页码，形式为 32,41,58 , 87--101 或者 68+。最后那一种形式表示 68 页及以后所有页。单个连字符表示页码范围，要被转化为标准样式中的双连字符形式，并生成一个小破折号，如 ‘87-101’。

publisher

出版社的名称。

school 写作论文时所在的学术机构的名称。

series 一系列书或一套书的名称。当从这个系列中引用一本书时，**title** 域给出书的题目，而可省的 **series** 给出整系列书的标题。

title 作品的题目，要遵从 B.2.4 节给出的大写规则。

type 技术报告的类型，例如 ‘Research Note’。

volume 期刊或者多卷书籍的卷号。

year 作品印刷的年代, 或者未出版作品的完成年代。年代通常要求有四位数, 例如 1999 。

另外还有一些域名, BibTeX 通常就把它忽略掉。例如, 要想向数据库中加入文章的摘要, 可以用

```
abstract = {摘要文本}
```

这对于我们的应用不只是管理这个数据库时, 就非常有用了。

§B.2.3 域的交叉引用

**** 本节内容只适用于 BibTeX 0.99 版本及以后版本。 ****

如果参考文献数据库中有许多条目有相同的域信息集合 (例如出现在同一会议论文集集中的几篇作品), 那么就可以用 **crossref** 域来引用另一个条目中的与自己相同的域信息。例如,

```
@INPROCEEDINGS{xyz-1,
  crossref = {xyz-proceedings},
  author = {J. S. Jones},
  title = {The First Results from the {Appleville Experiment}}
  pages = {34--38} }
. . . . .
@PROCEEDINGS{xyz-proceedings,
  editor = {C. H. Kelvin},
  title = {Proceedings of the First Conference on the
           {Appleville Experiment}},
  booktitle = {Proceedings of the First Conference on the
               {Appleville Experiment}}
  year = 1991 }
```

第一项关键词是 **xyz-1**, 利用 **crossref** 域从关键词为 **xyz-proceedings** 的第二个条目得到自己所缺少的域信息。这里缺少的域有 **editor**, **booktitle** 和 **year**, 这也是对会议论文集集中所有论文都一样的域。注意在 **@PROCEEDINGS** 中 **booktitle** 域是被忽略的, 而这儿必须包含它, 因为 **@INPROCEEDINGS** 需要这一个域的信息。

如果一个条目被其它两个或更多条目引用了, 即使没有 **\cite** 或 **\nocite** 命令用其关键词做参数值, 这项也会包含在参考文献中。

为了使整个系统工作正常, 在数据库中那些被引用的条目必须放在引用它们的条目后面。因此我们建议把所有这些只被其它条目引用的条目放在数据库尾部。交叉索引不能嵌套。

§B.2.4 特殊的域格式

对输入到 `author`, `editor`, `title` 和 `booktitle` 中的域文本必须遵从如下几条规则。当 `BIBTEX` 处理姓名时, 它会根据样式文件中的指令, 首先给出作者的姓, 然后是名的首字母缩写。因此如何告诉系统哪是名, 哪是姓就非常重要了。对标题也有类似的问题, 要根据样式和 / 或条目类型进行首字母大写, 因此 `BIBTEX` 必须知道哪些单词要被大写。

姓名

在 `author` 和 `editor` 域中的姓名既可以输入为 {名姓} 的格式, 也可以是 {姓, 名} 格式。也就是说, 如果没有逗号, `BIBTEX` 就认为第一个首字母大写的单词是姓; 否则就把逗号前面的文本当做姓。因此姓名文本 "John George Harrison" 和 "Harrison, John George" 都指的是 J. G. Harrison 先生。然而, 如果一个人有复姓, 中间又没有连字符隔开, 那就必须采用第二种方式, 或者复姓放在大括号内, 如

"San Martino, Maria" 或 "Maria {San Martino}"

都表示 M. San Martino 夫人。

对于姓中首字母没有大写的辅词, 例如 `von` 或 `de`, 则可以采用任一种方式输入:

"Richard von Mannheim" 或 "von Mannheim, Richard"

"Walter de la Maire" 或 "de la Maire, Walter"

放在大括号内的任意文本都当做一项处理, 这可以避免有时候会出现的歧义问题, 例如姓名中包含逗号或单词 `and`。

"{Harvey and Sons, Ltd}"

如果在姓名前要加上 `Junior`, 那会使事情变得更复杂。如果在 `Jr` 和名字之间有逗号, 则就应该写成如下形式:

"Ford, Jr, Henry"

然而如果是逗号, 那就必须把 `Jr` 看成是复姓的一部分:

"{Filmore Jr}, Charles" 或 "Charles {Filmore Jr}"

****`BIBTEX` 0.99 版本或以后版本 **** 姓名中的用反斜杠命令构成的重音应该放在大括号内, 并且反斜杠应该是紧接左大括号的第一个字符。通过这种方法, 字母排序和在 `alpha` 参考文献样式中对标签的格式化都会工作正常。例如,

```
author = "Kurt G{\\"o}del",
year = 1931
```

就会生成所希望的标签 [Göd31]。重音命令被大括号包围的深度不能超过这里所示例的。(在所有版本的 `BIBTEX` 中, 都可以把重音包含在姓名文本中,

其在参考文献中的显示是相当正确的。而这里新增加的功能就是对标签和字母排序的处理。)

****BIBTEX 0.99 版本及以后**** 名字中间有连字符, 也能正确地缩写。因此 "Jean-Paul Sartre" 成为 'J.-P. Sartre'。(在以前的版本中, 其结果为 'J. Sartre')。

如果 `author` 或者 `editor` 域中包含不只一个姓名, 那么两个姓名之间要用 `and` 分开。例如,

```
author = "Helmut Kopka and Daly, Patrick William" 或者
AUTHOR = {Peter C. Barnes and Tolman, Paul and Mary Smith}
```

如果 `and` 真的是姓名的一部分, 那么就必须把整个姓名放在大括号内, 见前面的示例。如果作者清单太长, 无法列出所有的姓名, 那么可以用 `and others` 表示结束。根据样式文件的规定, 这将会被转化为 *et al.*。

题目

是否对题目首字母进行大写, 与所用的参考文献样式有关: 通常书籍的题目首字母要大写, 而文章的题目则不这样做。在 `title` 和 `booktitle` 域中的文本应写成大写的形式, 这样必要时 BIBTEX 可以把它转化为小写形式。

在使用英语的国家, 大写题目的一般规则是: 题目的第一个单词, 冒号后的第一个单词, 以及其它单词中除冠词, 没有重音的介词及连词外都要首字母大写。例如,

```
title="The Right Way to Learn: A Short-Cut to a Successful Life"
```

那些总是首字母要大写的单词(例如专用名词)应该放在大括号内。实际上只要把总是要大写的字母放在大括号内就可以了。下面两个例子是等价的:

```
title = "The {Giotto} Mission to Comet {Halley}" 或
TITLE = {The {G}iotto Mission to Comet {H}alley}
```

§B.2.5 缩写

在输入域文本时通常可以使用缩写表示实际的文本。有些缩写, 例如月份名称和某些标准期刊名称就可以使用缩写, 当然用户也可以定义自己使用的缩写。

名称的缩写可以是字母、数字或符号的任意组合, 当然下面的符号除外:

```
" # % ' ( ) , = { }
```

缩写是用下面的命令定义的:

```
@string{缩写名称 = {文本}} 或
@string(缩写名称 = {文本})
```

其中 缩写名称 就是缩写后的形式，而 文本 是替换文本。例如，如果定义了下面这样的缩写：

```
@string{JGR = {Journal of Geophysical Research}}
```

那么下面这两条声明是一样的：

```
journal = JGR
```

```
journal = {Journal of Geophysical Research}
```

缩写名称不要放在大括号或双引号内部，否则就会把它按字面解释成域文本。

在 `@string` 命令和缩写名称中，都不区分大小写。因此上面的缩写也可以如下定义：

```
@STRING{jgr = {Journal of Geophysical Research}} 或者
```

```
@String{jGr = {Journal of Geophysical Research}}
```

在域中就可以用 JGR, JGr, JgR, Jgr, jGR, jGr, jgR 或者 jgr 来引用它。

****BibTeX 0.99 版本及以后**** 可以在缩写之间加上 `#` 来把它们结合在一起。因此如果定义了

```
@string{yrbk = {Institute Yearbook}}
```

那么这个缩写就可以同其它缩写或文本结合在一起，例如

```
title = "Max-Planck~" # yrbk # 1993
```

的结果为 ‘Max-Planck Institute Yearbook 1993’。

月份名称是有标准缩写的，其由原来名称的前三个字母组成，例如 jan, feb 等等。对于某些标准期刊，存在着一些预先定义的名称缩写。要想知道到底有哪些缩写，那就需要查询机房使用手册了，因为这与安装版本有关。

`@string` 命令可以放在数据库中任两项条目之间，但是必须是定义在使用之前。因此把所有的缩写定义放在数据库开头就是相当合理的。

§B.2.6 使用模板

向参考文献条目中输入文本似乎是一件相当复杂的工作，因为要记住那么多的事情，比如哪些域是不可少的，哪些域是可省的，它们的格式怎样，不一而足。简化这件工作的一种方法就是为经常使用的条目创建 模板。所谓模板，也就是一个基本完整的条目，只是所有的域都是空的。把模板存到单独一个文件中，只要想生成一项新条目，就把它插入到文件中。然后再输入域文本。

相应于 `@article` 条目类型的一个比较适当的模板可以是

```
@ARTICLE{<key>,
  AUTHOR   = {},
  TITLE    = {},
  YEAR     = {},
  JOURNAL  = {},
```

```

VOLUME = {},
NUMBER = {},
MONTH   = {},
PAGES   = {}
}

```

其中 `<key>` 是提醒用户这里必须要输入一个关键词，然后首先是不可少的域，并向里缩进，再后接可省域，并进一步缩进。这里没有包含域 `note`，因为虽然所有条目类型中都可以包含它，但很少会用到。

最后提一下，BibTeX是由Stanford大学的Oren Patashnik在Leslie Lamport的密切协助下开发的。安装了BibTeX后应该有这两个文件 `btldoc.tex` 和 `btldhak.tex`，其中就是一些关于BibTeX的信息。特别地，`btldhak.tex` 中有关于编写参考文献样式文件的指令。用 \LaTeX 处理这两个文件，就可以看到其中的内容了。

§B.3 扩展 BibTeX

§B.3.1 作者-年代 参考文献样式

本书中所有的参考文献样式同绝大多数的自然科学期刊一样，与 \LaTeX 所提供的标准样式不同。在这种样式中，对其它出版物的索引是通过引用作者姓名和出版年代进行的。而且，引用既可以是放在方括号内的，如 *[Jones et al., 1990]*，也可以是平铺直述的，如 *Jones et al. [1990]*。这种样式也可以有许多变体：例如，本书中是用圆括号取代了方括号，而且名称用的是直立字体。

在标准的 \LaTeX 中，引用是放在方括号内的一个活动编号，如 [10]，也允许逗号后面加上页码，如 [10, page 188]，但不能用文本引用。这里的编号也就是标签，并放到参考文献中，后接索引的文献信息。而 作者-年代 参考文献中没有标签，从而直接利用 作者 和 年代 可以容易知道引用与索引之间的关系。

实际上有许多非标准的作者-年代参考文献样式可以使用，所有这些样式都要利用一些宏包文件来重新设计 `thebibliography` 环境从而能解释修正后的 `\bibitem` 命令。

这其中最简单的一种样式就是由 BibTeX开发者 Oren Patashnik 所设计，称为 `apalike.bst` 和 `apalike.sty`。在这种样式中，`\bibitem` 有一个包含作者和年代的可省参数值，如

```
\bibitem[Jones et al., 1990]{jone90}
```

`\cite` 命令的使用与通常的一样，也就是只可以使用插入式的引用。至少有 8 个 `.bst` 文件遵从这种框架。

还有另外一组 L^AT_EX 和 B_IB_TE_X 样式, 称为 `newapa.sty` 和 `newapa.bst`, 由 Stephen N. Spence 开发出来, Young U. Ryu 对其进行了修改, 最后一次更新是在 1991 年 6 月进行的。这是一个具有相当弹性的系统, 引用既可以是插入式的, 也可以是平铺直述的, 而且作者的名单可以用全名, 也可以缩写。这时 `\bibitem` 命令的形式为

```
\bibitem[\protect\citeauthoryear{Jones, Smith, and Harris}
        {Jones et al.}{1990}]{jone90}
```

`\bibitem` 命令的可省参数中的特殊命令 `\citeauthoryear` 是用来以三个独立项的形式向 L^AT_EX 文档传递作者全名单, 缩写清单, 年代。这条命令是在相应的 `.sty` 宏包文件中定义的。它同时提供了大量的 `\cite` 命令变体, 用来显示名单与年代的组合, 这既可以是插入式的, 也可以是平铺直述的。有许多其它的参考文献样式, 其中有引人瞩目的 `chicago` 族, 也利用了这条命令。

而对于 `newapa`, 也可以很容易地改变引用中的标点符号, 比如说把 (Jones & Smith: 1990, pages 60–65) 改成 [Jones and Smith, 1990: pages 60–65]。这是利用 `\citepunct` 命令来做到的, 利用它可以得到任何所希望的标点符号用法。

在参考文献样式中还有一个叫 `astronomy` 的家族, 其中至少有 7 位成员。它们都应用了一条特殊命令 `\astroncite`, 它实际上是与前面提到的命令 `\citeauthoryear` 一致, 但它只有两个参数值, 分别相应于简短作者名单和年代。要用它们需要宏包 `astronomy.sty`。

另外一组属于 作者-年代 参考文献样式的应当是 `harvard` 族了。它们利用 `\harvarditem` 命令取代了 `\bibitem`, 但功能与 `\citeauthoryear` 一样。应用它们时, 需要由 Peter Williams 和 Thorsten Schnier 所写的宏包 `harvard.sty`。这个宏包已经进行了更新, 从而适用于 L^AT_EX 2_ε, 现在它包含了许多新功能来定制引用。其已成为了一个功能强大的宏包, 选用时, 相比上面所提到的其它宏包, 应当优先考虑这一个。

另外, 也有一些样式是在 `\bibitem` 的可省参数中应用命令 `\citename`。然而, 其语法与其它的不同, 而且据我们的看法, 其也没有涵盖所有可能的引用形式。

Patrick W. Daly 则完成了一个更一般的宏包, 它接受上面所有 `\bibitem` 中 作者-年代 格式, 也接受数字引用框架。这个宏包名称为 `natbib.sty`, 它实际上是把年代在圆括号内的可省参数取作 作者-年代 的基本形式,

```
\bibitem[Jones et al.(1990)]{jone90}
```

并用这种格式定义了所有其它 作者-年代 命令。而且也可以把所有括号和标点记号的类型, 以及其它所需要的特殊功能, 与 `.bst` 文件的名称联系起来, 这样当调用 `\bibliography` 命令时就会调用它。因此所有的使用同样参考文

献样式的文档都会拥有正确的引用标点符号，甚至不用对作者部分做任何其它修改。而且对数字和 作者-年代 样式的选择是完全自动的。这个宏包已经相应于 L^AT_EX 2_ε进行了更新，拥有与 `harvard.sty` 同样的功能，因此可推荐为所有参考文献样式的普适性宏包。

在 C.3.4 节给出了 `natbib.sty` 的一个简化版本，并同时给出一个基本的作者-年代 参考文献样式。

所有在这里提到的样式，包括 `natbib.sty` 在内，都可以从 CTAN 文件服务器上的 BibTeX 和 L^AT_EX 相应目录中取到，我们在 D.5 列出了这些服务器，

§B.3.2 定制参考文献样式

好像每份期刊和出版社都有自己的参考文献格式规则。这其中的差异实际上是微乎其微的，比如说此处应该用逗号还是冒号，或者编号字体为黑体还是斜体。但是，虽然差异很小，每个出版社对格式的要求却是非常严格的。

标准的 L^AT_EX 只提供了四种参考文献样式，它们的差别也就只限于排序和标签等琐碎的细节。我们可以通过修改已有的样式文件，来设计自己的 `.bst` 文件；然而，这需要对独特的 BibTeX 程序设计语法有相当的了解，因为这种语法甚至会使有经验的 L^AT_EX 用户也感到迷糊。在 T_EX 文件服务器上的 BibTeX 目录中大约有 50 个 `.bst` 文件，而在 L^AT_EX 相关的目录中可能有更多的此类文件。其中每个文件都是专对于某一期刊而设计的，因此无法保证能被另外的期刊所接受。如果出版社要求的格式，在那些地方却找不到，该怎么办呢？而且又应怎样在这些格式中搜索自己想要的格式呢？

从 `custom-bib` 宏包中可以得到一些帮助，这个宏包是由 Patrick W. Daly 写成，可以从文件服务器上的 L^AT_EX 相关目录中找到。这个宏包的核心就是一个通用的（或者说主要的）参考文献样式 `.mbs`，其中包含了相当多的针对于不同参考文献功能或选项的替换代码。要用 DocStrip 程序处理它，这个程序能根据给定的选项来包含进来替换代码（见 D.3.1 节）。

由于提供了大量的选项（约有 100 个），因此设计了一个菜单驱动界面，称为 `makebst.tex`。当用 T_EX 或 L^AT_EX 处理这个“程序”时，它就会首先问你要读入哪个 `.mbs` 文件，接着就会交互构造出一个 DocStrip 批处理作业，以根据包含在 `.mbs` 文件中的菜单信息生成所选择的参考文献特征。这也就是说可以有很多 `.mbs` 文件供选择，每一个都通过 `makebst` 向用户解释自己的选项集合。

事实上，`merlin.mbs` 是第三代公开发表的参考文献样式，它取代了原来的 `genbst.mbs` 以及多语言部分 `babel.mbs`。对其它语言的支持现在是通过另外的 `.mbs` 文件提供的，每个相应于一种语言，其中包含对类似于 *volume*, *editor*, *edition* 等等单词的翻译。

由通用参考文献样式文件所提供的一些功能如下：

- 数字或 作者-年代 引用；对后者，用户可以选择所使用的 `\bibitem` 样式；
- 索引的顺序：排序可以按引用顺序，也可以按作者姓名的字母顺序，或者根据作者 - 年代标签字母顺序；
- 作者姓名的格式：名加姓，名首字母加姓，姓加名首字母，只把第一作者的首字母颠倒过来，等等。
- 在 `et al.` 前面可以给出的作者姓名个数；
- 排版作者姓名的字样；
- 日期的位置，是否把年份放在圆括号或方括号内；
- 期刊的卷，期以及页码的格式；
- 以句子或者标题样式对文章标题进行大写；
- 在单词间用 *and* 还是用 *&*；
- 用逗号取代单词 *and* ；
- 是否缩写 *editor*, *volume*, *chapter* 等；
- 给出起始，结束页，还是只给出开始页；
- 常见期刊名称的简写形式；
-