

**APPENDIX B**

# **The Bibliography Database**



will be incorrect if any of these numbers have changed. L<sup>A</sup>T<sub>E</sub>X will warn you if this may have happened, in which case you should run it again on the input file to make sure the cross-references are correct. (This warning will occur if any number assigned to a key by a `\label` command has changed, even if that number is not referenced.) Each `\ref` or `\pageref` referring to an unknown key produces a warning message; such messages appear the first time you process any file containing these commands.

A `\label` can appear in the argument of a sectioning or `\caption` command, but in no other moving argument.

Using keys for cross-referencing saves you from keeping track of the actual numbers, but it requires you to remember the keys. You can produce a list of the keys by running L<sup>A</sup>T<sub>E</sub>X on the input file `lablst.tex`. (You probably do this by typing “`latex lablst`”; check your *Local Guide* to be sure.) L<sup>A</sup>T<sub>E</sub>X will then ask you to type in the name of the input file whose keys you want listed, as well as the name of the document class specified by that file’s `\documentclass` command.

## 4.3 Bibliography and Citation

A citation is a cross-reference to another publication, such as a journal article, called the *source*. The modern method of citing a source is with a cross-reference to an entry in a list of sources at the end of the document. With L<sup>A</sup>T<sub>E</sub>X, the citation is produced by a `\cite` command having the citation key as its argument.

Knudson [67] showed that, in the Arctic ...      Knudson<sup>~</sup>`\cite{kn:gнус}` showed ...

You can cite multiple sources with a single `\cite`, separating the keys by commas. The `\cite` command has an optional argument that adds a note to the citation.

Although they had disappeared from Fiji [4,15,36],      ... Fiji<sup>~</sup>`\cite{tom-ix,dick:ens,harry+d}`,  
Knudson [67, pages 222–333] showed that ...      Knudson<sup>~</sup>`\cite[pages 222--333]{kn:gнус}` ...

A citation key can be any sequence of letters, digits, and punctuation characters, except that it may not contain a comma. As usual in L<sup>A</sup>T<sub>E</sub>X, upper- and lowercase letters are considered to be different.

In the preceding examples, L<sup>A</sup>T<sub>E</sub>X has to determine that citation key `kn:gнус` corresponds to source label 67. How L<sup>A</sup>T<sub>E</sub>X does this depends on how you produce the list of sources. The best way to produce the source list is with a separate program called B<sub>I</sub>B<sub>T</sub><sub>E</sub>X, described in Section 4.3.1. You can also produce it yourself, as explained in Section 4.3.2.

### 4.3.1 Using BIBTEX

BIBTEX is a separate program that produces the source list for a document, obtaining the information from a bibliographic database. To use BIBTEX, you must include in your L<sup>A</sup>T<sub>E</sub>X input file a `\bibliography` command whose argument specifies one or more files that contain the database. The names of the database files must have the extension `bib`. For example, the command

```
\bibliography{insect,animal}
```

specifies that the source list is to be obtained from entries in the files `insect.bib` and `animal.bib`. There must be no space following the comma. Appendix B explains how to make bibliographic database files.

BIBTEX creates a source list containing entries for all the citation keys specified by `\cite` commands. The data for the source list is obtained from the bibliographic database, which must have an entry for every citation key. A `\nocite` command in the L<sup>A</sup>T<sub>E</sub>X input file causes the specified entries to appear in the source list, but produces no output. For example, `\nocite{g:nu,g:nat}` causes BIBTEX to put bibliography database entries having keys `g:nu` and `g:nat` in the source list. The command `\nocite{*}` causes all entries in the bibliographic database to be put in the source list. A `\nocite` command can go anywhere after the `\begin{document}` command, but it is fragile.

To use BIBTEX, your L<sup>A</sup>T<sub>E</sub>X input file must contain a `\bibliographystyle` command. This command specifies the *bibliography style*, which determines the format of the source list. For example, the command

```
\bibliographystyle{plain}
```

specifies that entries should be formatted as specified by the `plain` bibliography style. The `\bibliographystyle` command can go anywhere after the `\begin{document}` command. L<sup>A</sup>T<sub>E</sub>X's standard bibliography styles are:

**plain** Formatted more or less as suggested by van Leunen in *A Handbook for Scholars* [7]. Entries are sorted alphabetically and are labeled with numbers.

**unsrt** The same as `plain` except that entries appear in the order of their first citation.

**alpha** The same as `plain` except that source labels like "Knu66", formed from the author's name and the year of publication, are used.

**abbrv** The same as `plain` except that entries are more compact because first names, month names, and journal names are abbreviated.

Dozens of other bibliography styles exist, including ones that produce source lists in the formats used by a number of scientific journals. Consult the L<sup>A</sup>T<sub>E</sub>X

*Companion* and the *Local Guide* to find out what styles are available. Documentation for the `BIBTEX` program explains how to create your own bibliography style.

The source list is normally formatted in what van Leunen calls a *compressed* style. The `openbib` document-class option causes it to be formatted in an *open* style. (Document-class options are specified by the `\documentclass` command; see Section 2.2.2.)

Once you've created an input file containing the appropriate `LATEX` commands, you perform the following sequence of steps to produce the final output:

- Run `LATEX` on the input file, which I assume is called `myfile.tex`. `LATEX` will complain that all your citations are undefined, since there is no source list yet.
- Run `BIBTEX` by typing something like `bibtex myfile`. (Consult your *Local Guide* to find out what you actually type.) `BIBTEX` will generate the file `myfile.bbl` containing `LATEX` commands to produce the source list.
- Run `LATEX` again on `myfile.tex`. `LATEX`'s output will now contain the source list. However, `LATEX` will still complain that your citations are undefined, since the output produced by a `\cite` command is based on information obtained from the source list the last time `LATEX` was run on the file.
- Run `LATEX` one more time on `myfile.tex`.

If you add or remove a citation, you will have to go through this whole procedure again to get the citation labels and source list right. But they don't have to be right while you're writing, so you needn't do this very often.

`BIBTEX` almost always produces a perfectly fine source list. However, it is only a computer program, so you may occasionally encounter a source that it does not handle properly. When this happens, you can usually correct the problem by modifying the bibliographic database—perhaps creating a special database entry just for this document. As a last resort, you can edit the `bbl` file that `BIBTEX` generated. (Of course, you should do this only when you are producing the final output.)

### 4.3.2 Doing It Yourself

A source list is created with the `thebibliography` environment, which is like the `enumerate` environment described in Section 2.2.4 except that:

- Each list item is begun with a `\bibitem` command whose argument is the citation key. (The `\bibitem` and `\cite` commands work much like the `\label` and `\ref` commands of Section 4.2.)

As explained in Section 4.3.1, the `\bibliography` command specifies one or more `bib` files—bibliographic database files whose names have the extension `bib`. `BIBTEX` uses the `bib` file(s) to generate a `bbl` file that is read by `\bibliography` to make the bibliography. This appendix explains how to create `bib` files.

Once you learn to use `BIBTEX`, you will find it easier to let `BIBTEX` make your reference list than to do it yourself. Moreover, you will quickly compile a bibliographic database that eliminates almost all the work of making a bibliography. Other people may have `bib` files that you can copy, or there may be a common database that you can use. Ask your friends or check the *Local Guide* to find out what is available. However, remember that you are responsible for the accuracy of the references in your document. Even published references are notoriously unreliable; don't rely on any bibliography information that has not been carefully checked by someone you trust.

## B.1 The Format of the `bib` File

### B.1.1 The Entry Format

A `bib` file contains a series of entries like the following:

```
@BOOK{kn:gnus,
  AUTHOR = "Donald E. Knudson",
  TITLE  = "1966 World Gnus Almanac",
  PUBLISHER = {Permafrost Press},
  ADDRESS = {Novosibirsk} }
```

The `@BOOK` states that this is an entry of type *book*. Various entry types are described below. The `kn:gnus` is the *key*, as it appears in the argument of a `\cite` command referring to the entry.

This entry has four *fields*, named `AUTHOR`, `TITLE`, `PUBLISHER`, and `ADDRESS`. The meanings of these and other fields are described below. A field consists of the name, followed by an "=" character with optional space around it, followed by its text. The text of a field is a string of characters, with no unmatched braces, surrounded by either a pair of braces or a pair of " characters. (Unlike in `LATEX` input, `\{` and `\}` are considered to be braces with respect to brace matching.) Entry fields are separated from one another, and from the key, by commas. A comma may have optional space around it.

The outermost braces that surround the entire entry may be replaced by parentheses. As in `LATEX` input files, an end-of-line character counts as a space and one space is equivalent to one hundred. Unlike `LATEX`, `BIBTEX` ignores the case of letters in the entry type, key, and field names, so the entry above could have been typed as follows:

```
@Book(KN:Gnus, author={Donald E. Knudson} ,
      TITLE =      "1966 World
                   Gnus Almanac", ... )
```

However, the case of letters does matter to  $\text{\LaTeX}$ , so the key should appear exactly the same in all  $\text{\cite}$  commands in the  $\text{\LaTeX}$  input file.

The quotes or braces can be omitted around text consisting entirely of numerals. The following two fields are equivalent:

```
Volume = "27"           Volume = 27
```

### B.1.2 The Text of a Field

The text of the field is enclosed in braces or double quote characters ( $\text{\textquotesingle}$ ). A part of the text is said to be *enclosed in braces* if it lies inside a matching pair of braces other than the ones enclosing the entire field.

#### Names

The text of an `author` or `editor` field represents a list of names. The bibliography style determines the format in which a name is printed—whether the first name or last name appears first, if the full first name or just the first initial is used, etc. The `bib` file entry simply tells  $\text{\BIBTeX}$  what the name is. You should type an author's complete name, exactly as it appears in the cited work, and let the bibliography style decide what to abbreviate.

Most names can be entered in the obvious way, either with or without a comma, as in the following examples.

```
"John Paul Jones"           "Jones, John Paul"
"Ludwig van Beethoven"      "van Beethoven, Ludwig"
```

Only the second form, with a comma, should be used for people who have last names with multiple parts that are capitalized. For example, Per Brinch Hansen's last name is Brinch Hansen, so his name should be typed with a comma:

```
"Brinch Hansen, Per"
```

If you type `"Per Brinch Hansen"`,  $\text{\BIBTeX}$  will think that "Brinch" is his middle name. "van Beethoven" or "de la Madrid" pose no problem because "van" and "de la" are not capitalized.

"Juniors" present a special problem. People with "Jr." in their name generally precede it with a comma. Such a name should be entered as follows:

```
"Ford, Jr., Henry"
```

BIB<sub>T</sub>E<sub>X</sub> is sometimes confused by characters that are produced by L<sub>A</sub>T<sub>E</sub>X commands—for example, accented characters and characters produced by the commands of Section 3.2.2. It will do the right thing if you put curly braces immediately around a command that produces a character:

```
"Kurt G{\o}del"    "V. S{\o}rensen"    "J. Mart{\'}{\i}"
```

If there are multiple authors or editors, their names are separated by the word *and*. A paper written by Alpher, Bethe, and Gamow has the following entry:

```
AUTHOR = "Ralph Alpher and Bethe, Hans and George Gamow"
```

An *and* separates authors' names only if it is not enclosed in braces. If an *author* or *editor* field has more names than you want to type, just end the list of names with *and others*; the standard styles convert this to the conventional "et al."

### Titles

The bibliography style determines whether or not a title is capitalized; the titles of books usually are, the titles of articles usually are not. You type a title the way it should appear if it is capitalized.

```
TITLE = "The Agony and the Ecstasy"
```

You should capitalize the first word of the title, the first word after a colon, and all other words except articles and unstressed conjunctions and prepositions. BIB<sub>T</sub>E<sub>X</sub> will change uppercase letters to lowercase if appropriate. Uppercase letters that should not be changed are enclosed in braces. The following two titles are equivalent; the *A* of *Africa* will not be made lowercase.

```
"The Gnats and Gnus of {Africa}"
"The Gnats and Gnus of {A}frica"
```

### B.1.3 Abbreviations

Instead of an ordinary text string, the text of a field can be replaced by an *abbreviation* for it. An abbreviation is a string of characters that starts with a letter and does not contain a space or any of the following ten characters:

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

The abbreviation is typed in place of the text field, with no braces or quotation marks. If *jgg1* is an abbreviation for

```
Journal of Gnats and Gnus, Series~1
```



then the following are equivalent:

```
Journal = jgg1
Journal = "Journal of Gnats and Gnus, Series~1"
```

Some abbreviations are predefined by the bibliography style. These always include the usual three-letter abbreviations for the month: `jan`, `feb`, `mar`, etc. Bibliography styles may contain abbreviations for the names of commonly referenced journals. Consult your *Local Guide* for a list of the predefined abbreviations for the bibliography styles available on your computer.

You can define your own abbreviations by putting a `@string` command in the `bib` file. The command

```
@string{jgg1 = "Journal of Gnats and Gnus, Series~1"}
```

defines `jgg1` to be the abbreviation assumed in the previous example. Parentheses can be used in place of the outermost braces in the `@string` command, and braces can be used instead of the quotation marks. The text must have matching braces.

The case of letters is ignored in an abbreviation as well as in the command name `@string`, so the command above is equivalent to

```
@STRING{JgG1 = "Journal of Gnats and Gnus, Series~1"}
```

A `@string` command can appear anywhere before or between entries in a `bib` file. However, it must come before any use of the abbreviation, so a sensible place for `@string` commands is at the beginning of the file. You can also put your abbreviations in a separate `bib` file, say `abbrev.bib`, and use the command

```
\bibliography{abbrev,...}
```

in your document. A `@string` command in a `bib` file takes precedence over a definition made by the bibliography style, so it can be used to change the definition of an abbreviation such as `Feb`.

#### B.1.4 Cross-References

Several cited sources may be part of a larger work—for example, different papers in the same conference proceedings. You can make a single entry for the conference proceedings, and refer to that entry in the entries for the individual papers. Fields that appear in the proceedings' entry need not be duplicated in the papers' entries. However, every required field for a paper must be either in its entry or in the referenced entry.



```

@INPROCEEDINGS(beestly-gnats,
  AUTHOR   = "Will D. Beest",
  TITLE    = "Gnats in the Gnus",
  PAGES    = "47--59",
  CROSSREF = "ope:6cpb")
...
@PROCEEDINGS(ope:6cpb,
  TITLE     = "Sixth Conference on Parasites in Bovidae",
  BOOKTITLE = "Sixth Conference on Parasites in Bovidae",
  EDITOR    = "Ann T. L. Ope",
  YEAR      = 1975)

```

The apparently redundant `BOOKTITLE` field in the proceedings entry is needed to provide the field of that name for the entry of each paper that cross-references it. As explained below, the `TITLE` field is required to produce a reference-list entry for the proceedings; `BIBTEX` ignores the `BOOKTITLE` field when producing such an entry. The reference list made by `BIBTEX` may have an entry for the proceedings that is cited by the entries for the individual papers, even if the proceedings are not explicitly cited in the original document.

A cross-referenced entry like `ope:6cpb` in the example must come after any entries that refer to it.

## B.2 The Entries

### B.2.1 Entry Types

When entering a reference in the database, the first thing to decide is what type of entry it is. No fixed classification scheme can be complete, but `BIBTEX` provides enough entry types to handle almost any reference reasonably well.

References to different types of publications contain different information; a reference to a journal article might include the volume and number of the journal, which is usually not meaningful for a book. Therefore, database entries of different types have different fields. For each entry type, the fields are divided into three classes:

**required** Omitting the field will produce an error message and will occasionally result in a badly formatted bibliography entry. If the required information is not meaningful, you are using the wrong entry type. If the required information is meaningful but not needed—for example, because it is included in some other field—simply ignore the warning that `BIBTEX` generates.

**optional** The field's information will be used if present, but can be omitted without causing any formatting problems. A reference should contain any information that might help the reader, so you should include the optional

field if it is applicable. (A nonstandard bibliography style might ignore an optional field when creating the reference-list entry.)

**ignored** The field is ignored. `BIBTEX` ignores a field that is not required or optional, so you can include any fields you want in a `bib` file entry. It's a good idea to put all relevant information about a reference in its `bib` file entry—even information that may never appear in the bibliography. For example, if you want to keep an abstract of a paper in a computer file, put it in an `abstract` field in the paper's `bib` file entry. The `bib` file is likely to be as good a place as any for the abstract, and it is possible to design a bibliography style for printing selected abstracts.

Misspelling its name will cause a field to be ignored, so check the database entry if relevant information that you think is there does not appear in the reference-list entry.

The following are all the entry types, along with their required and optional fields, that are used by the standard bibliography styles. The meanings of the individual fields are explained in the next section. A particular bibliography style may ignore some optional fields in creating the reference. Remember that, when used in the `bib` file, the entry-type name is preceded by an `@` character.

**article** An article from a journal or magazine. Required fields: `author`, `title`, `journal`, `year`. Optional fields: `volume`, `number`, `pages`, `month`, `note`.

**book** A book with an explicit publisher. Required fields: `author` or `editor`, `title`, `publisher`, `year`. Optional fields: `volume` or `number`, `series`, `address`, `edition`, `month`, `note`.

**booklet** A work that is printed and bound, but without a named publisher or sponsoring institution. Required field: `title`. Optional fields: `author`, `howpublished`, `address`, `month`, `year`, `note`.

**conference** The same as **inproceedings**, included for compatibility with older versions.

**inbook** A part of a book, usually untitled; it may be a chapter (or other sectional unit) and/or a range of pages. Required fields: `author` or `editor`, `title`, `chapter` and/or `pages`, `publisher`, `year`. Optional fields: `volume` or `number`, `series`, `type`, `address`, `edition`, `month`, `note`.

**incollection** A part of a book with its own title. Required fields: `author`, `title`, `booktitle`, `publisher`, `year`. Optional fields: `editor`, `volume` or `number`, `series`, `type`, `chapter`, `pages`, `address`, `edition`, `month`, `note`.

**inproceedings** An article in a conference proceedings. Required fields: `author`, `title`, `booktitle`, `year`. Optional fields: `editor`, `volume` or `number`, `series`, `pages`, `address`, `month`, `organization`, `publisher`, `note`.

**manual** Technical documentation. Required field: `title`. Optional fields: `author`, `organization`, `address`, `edition`, `month`, `year`, `note`.

**mastersthesis** A master's thesis. Required fields: `author`, `title`, `school`, `year`. Optional fields: `type`, `address`, `month`, `note`.

**misc** Use this type when nothing else fits. Required fields: none. Optional fields: `author`, `title`, `howpublished`, `month`, `year`, `note`.

**phdthesis** A Ph.D. thesis. Required fields: `author`, `title`, `school`, `year`. Optional fields: `type`, `address`, `month`, `note`.

**proceedings** The proceedings of a conference. Required fields: `title`, `year`. Optional fields: `editor`, `volume` or `number`, `series`, `address`, `month`, `organization`, `publisher`, `note`.

**techreport** A report published by a school or other institution, usually numbered within a series. Required fields: `author`, `title`, `institution`, `year`. Optional fields: `type`, `number`, `address`, `month`, `note`.

**unpublished** A document with an author and title, but not formally published. Required fields: `author`, `title`, `note`. Optional fields: `month`, `year`.

In addition to the fields listed above, each entry type also has an optional `key` field, used in some styles for alphabetizing and forming a `\bibitem` label. You should include a `key` field for any entry with no `author` or `author substitute`. (Depending on the entry type, an `editor` or an `organization` can substitute for an `author`.) Do not confuse the `key` field with the key that appears in the `\cite` command and at the beginning of the whole entry, after the entry type.

## B.2.2 Fields

Below is a description of all the fields recognized by the standard bibliography styles. An entry can also contain other fields that are ignored by those styles.

**address** Usually the address of the `publisher` or `institution`. For major publishing houses, omit it entirely or just give the city. For small publishers, you can help the reader by giving the complete address.

**annote** An annotation. It is not used by the standard bibliography styles, but may be used by other styles that produce an annotated bibliography.

**author** The name(s) of the author(s), in the format described above.

**booktitle** The title of a book, a titled part of which is being cited. It is used only for the `incollection` and `inproceedings` entry types; use the `title` field for `book` entries. How to type titles is explained above.

- chapter** A chapter (or other sectional unit) number.
- crossref** The database key of the entry being cross-referenced.
- edition** The edition of a book—for example, “Second”. (The style will convert this to “second” if appropriate.)
- editor** The name(s) of editor(s), typed as indicated above. If there is also an **author** field, then the **editor** field gives the editor of the book or collection in which the reference appears.
- howpublished** How something strange was published.
- institution** The sponsoring institution of a technical report.
- journal** A journal name. Abbreviations may exist; see the *Local Guide*.
- key** Used for alphabetizing and creating a label when the **author** and **editor** fields are missing. This field should not be confused with the key that appears in the `\cite` command and at the beginning of the entry.
- month** The month in which the work was published or, for an unpublished work, in which it was written. Use the standard three-letter abbreviations described above.
- note** Any additional information that can help the reader. The first word should be capitalized.
- number** The number of a journal, magazine, technical report, or work in a series. An issue of a journal or magazine is usually identified by its volume and number; the organization that issues a technical report usually gives it a number; books in a named series are sometimes numbered.
- organization** The organization that sponsors a conference or that publishes a manual.
- pages** One or more page numbers or ranges of numbers, such as 42--111 or 7, 41, 73--97.
- publisher** The publisher’s name.
- school** The name of the school where a thesis was written.
- series** The name of a series or set of books. When citing an entire book, the **title** field gives its title and the optional **series** field gives the name of a series or multivolume set in which the book was published.
- title** The work’s title, typed as explained above.

**type** The type of a technical report—for example, “Research Note”. It is also used to specify a type of sectional unit in an **inbook** or **incollection** entry and a different type of thesis in a **mastersthesis** or **phdthesis** entry.

**volume** The volume of a journal or multivolume book.

**year** The year of publication or, for an unpublished work, the year it was written. It usually consists only of numerals, such as 1984, but it could also be something like **circa 1066**.