

BIB_{TEX} yesterday, today, and tomorrow

Oren Patashnik
San Diego, CA

Abstract

This paper looks back at the last 20 years of BIB_{TEX}, and also looks ahead, if not to the next 20 years, then at least to BIB_{TEX} 1.0.

Introduction

BIB_{TEX} is the bibliography program designed originally to accompany Leslie Lamport's L^A_{TEX}; it now works with other incarnations of T_{EX}, too. BIB_{TEX} removes the tedium, and adds some flexibility, in producing a reference list.¹ When BIB_{TEX} creates your reference list, it's BIB_{TEX}, not you, minding the minutiae like ensuring that your reference-list entries are in the correct order, that every comma is in place, and that the information is formatted consistently across entries. Furthermore, a single, simple, change of bibliography-style name lets you convert your reference list from style A, which might order the entries alphabetically, and spell out journal names in full, and list all authors as first-name then last-name, to a completely different style B, which might order the entries according to their order of mention in the text, and abbreviate journal names, and invert just the first author's first and last names.

This paper is an updated version of a paper [7] from the 1994 TUG meeting in Santa Barbara. It gives: a history of BIB_{TEX} to the present; the general goals for BIB_{TEX} 1.0, which will be the frozen version of BIB_{TEX} (just as T_{EX} 3 is the frozen version of T_{EX}); some specific new features for achieving those goals; and the plan for releases of BIB_{TEX} leading up to BIB_{TEX} 1.0. Before all that, however, primarily for those who are unfamiliar with BIB_{TEX}, comes a section that explains its rudiments.

Using BIB_{TEX}

To use BIB_{TEX}, you'll have your bibliographic information in a bibliography database, and, to make use of that information, a few L^A_{TEX}² commands sprinkled throughout your L^A_{TEX} source file.

¹ Throughout this paper, the term 'reference list' is used generally to refer to what might also be called a 'bibliography' or a 'list of sources' or anything similar.

² The term 'L^A_{TEX}' is used to mean either L^A_{TEX} or plain (or other variations of) T_{EX}.

For example in a file `mybib.bib` (database file names end with `.bib`) you might have an entry like:

```
@BOOK{knuth:tex,
  author = "Donald E. Knuth",
  title = "The {\TeX}book",
  publisher = "Addison-Wesley",
  year = "1984",
}
```

The `@BOOK` tells BIB_{TEX} the entry type. (The bibliography style will instruct BIB_{TEX} on how to format a `BOOK` entry type.) The `knuth:tex` is the database key, which is a sequence of characters to be used as the symbolic name for this entry. And the rest of the entry comprises four `<field> = <field-value>` pairs appropriate for a `BOOK` entry type. In general you will have many such entries in a database file; you might also have multiple database files.

And in your L^A_{TEX} source file you might have a citation like

```
... in the \TeX{}book~\cite{knuth:tex} ...
```

The `\cite` command's argument `knuth:tex`, called a cite key, must match the corresponding database key. L^A_{TEX} might typeset this `\cite` command as

```
... in the TEXbook [41] ...
```

or

```
... in the TEXbook41 ...
```

or

```
... in the TEXbook (Knuth, 1984) ...
```

depending on the citation style. L^A_{TEX}'s default citation style uses a number in brackets, and for that citation style, together with an appropriate bibliography style, the corresponding reference-list entry might look like:

41. Donald E. Knuth. *The T_{EX}book*. Addison-Wesley, 1984.

Besides the `\cite` commands, your L^A_{TEX} source file will also have two BIB_{TEX}-related commands:

```
\bibliography{mybib}
\bibliographystyle{plain}
```

The `\bibliography` command does two things: It tells \LaTeX to put the reference list at that spot in your document, and it tells $\text{BIB}\TeX$ which file(s) to use for the bibliographic database, here just the single file `mybib.bib`. The `\bibliographystyle` command tells \LaTeX nothing, but tells $\text{BIB}\TeX$ which bibliography style to use, here the standard style `plain`; bibliography style file names end with `.bst`, thus the relevant file is `plain.bst` in this case.

So with your database file(s) and your \LaTeX source file structured appropriately, your citations are formatted according to the citation style, and your reference list is formatted according to the bibliography style.

To actually produce the typeset document, you run \LaTeX , $\text{BIB}\TeX$, \LaTeX , \LaTeX . The first \LaTeX run writes, to an `.aux` file, information for use by $\text{BIB}\TeX$ — which bibliography style to use, which database file(s) to use, and which database entries to include. The $\text{BIB}\TeX$ run reads all that information from the `.aux` file, reads the specified database (`.bib`) file(s), formats the reference list according to the instructions in the bibliography style (`.bst`) file, and writes its output onto a `.bbl` file. The next \LaTeX run reads the `.bbl` file and incorporates the reference list into the document. The final \LaTeX run fixes the references into the reference list. Figure 1 shows the files that $\text{BIB}\TeX$ uses. The `.blg` file is $\text{BIB}\TeX$'s log file, in which $\text{BIB}\TeX$ records any warning or error messages.

To try using $\text{BIB}\TeX$ with \LaTeX , put the six-line `BOOK` entry shown on the previous page into a file called `mybib.bib`, and then, into a file called `mypaper1.tex`, put these six lines of \LaTeX :

```
\documentclass{article}
\begin{document}
The \TeX{}book~\cite{knuth:tex} is good.
\bibliography{mybib}
\bibliographystyle{plain}
\end{document}
```

Exactly how you run \LaTeX and $\text{BIB}\TeX$ is system dependent; on my system I issue four commands:

```
latex mypaper1
bibtex mypaper1
latex mypaper1
latex mypaper1
```

To try using $\text{BIB}\TeX$ with plain \TeX , create the file `mybib.bib` as above, and then put into a file called `mypaper2.tex` these seven lines of plain \TeX :

```
\input btxmac
The \TeX{}book~\cite{knuth:tex} is good.
\medskip
\leftline{\bf References}
\bibliography{mybib}
```

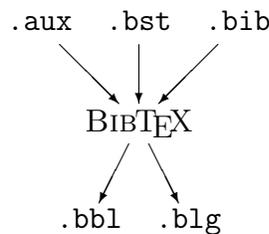


Figure 1: $\text{BIB}\TeX$'s input and output files.

```
\bibliographystyle{plain}
\bye
```

To run `mypaper2` through \TeX and $\text{BIB}\TeX$:

```
tex mypaper2
bibtex mypaper2
tex mypaper2
tex mypaper2
```

The file `btxmac.tex`, which `mypaper2.tex` `\inputs`, contains the macros that make $\text{BIB}\TeX$ work with plain \TeX . Those macros are a standard part of most \TeX distributions, but if they're not a part of yours, you'll have to go fetch a copy.

That's a brief introduction to $\text{BIB}\TeX$. The following sources provide further details. Leslie Lamport's \LaTeX manual [4] explains how to use $\text{BIB}\TeX$ with \LaTeX . In particular, section B.1 describes the `.bib`-file format in detail. The file `btxmac.tex` [1] documents its own use, with or without Karl Berry's `eplain.tex` package (for which the `btxmac` macros were originally written). The “ $\text{BIB}\TeX$ ing” document [5], which is distributed along with $\text{BIB}\TeX$ itself, contains further hints for $\text{BIB}\TeX$ users. The “Designing $\text{BIB}\TeX$ Styles” document [6], also distributed with $\text{BIB}\TeX$, explains the postfix stack-based language used to write $\text{BIB}\TeX$ bibliography styles (`.bst`) files. *The \LaTeX Companion* (2nd edition) [2], by Michel Goossens, Frank Mittelbach, et al., and *Guide to \LaTeX* [3], by Helmut Kopka and Patrick W. Daly, summarize much of the information in the sources above, and describe some of the tools available for helping with $\text{BIB}\TeX$. $\text{BIB}\TeX$'s standard bibliography styles, like `plain`, are based on Mary-Claire van Leunen's *A Handbook for Scholars* [9]. That book is worthwhile reading for anyone wanting to design a bibliography style from scratch.

History

Leslie Lamport patterned \LaTeX after a document production system called *Scribe* [8], written by Brian Reid in the late 1970s at Carnegie-Mellon University. One of *Scribe*'s basic tenets was that, to the extent possible with a computer program, writers should be allowed to concentrate on content rather

than on formatting details. Or, as Reid so amusingly put it:³

Not everyone should be a typesetter.

I think of L^AT_EX as a fairly successful Scribification of T_EX — L^AT_EX is almost as easy to use as Scribe yet almost as powerful as T_EX.

In any case, Scribe had become popular in certain academic circles, and Lamport decided that, to make it easy for Scribe users to convert to L^AT_EX, he would adopt Scribe’s bibliography scheme in L^AT_EX. But T_EX macros alone were insufficient in practice to do all the things, like alphabetizing, that a bibliography processor needs to do; he decided instead to have a separate bibliography program. That program would manipulate the bibliographic information in Scribe-like database files according to the instructions programmed in a special-purpose style language. The postfix stack-based language he had in mind was to be powerful enough to program many different bibliography styles.

My own work on BIB_TE_X started in February 1983. Just for the fun of it, I went back and dug up the original email:

```
15-Feb-83 0908
To: OP@SU-AI, tex82@SRI-CSL
Oren,
Leslie Lamport is working on a new
macro package for TeX82, and needs
someone to write a support program
or two. We are volunteering you. His
address is TEX82@SRI-CSL. Why don't
you get in contact with him, and see
whether his requirements are within
the scope of what you're willing to do.
```

This was described as a “three-week project”, and the first time I gave this talk, at Santa Barbara in 1994, I compared the “three-week project” to the “three-hour tour” of the 1960’s American television series *Gilligan’s Island*, in which an afternoon’s harbor cruise became a shipwreck adventure lasting years. (Many of us at that TUG conference really did go on a three-hour harbor cruise.)

This year’s meeting was on the Big Island of Hawaii, though, and it struck me that there’s an even better metaphor for my work on BIB_TE_X than the Gilligan’s Island adventure: I started working on BIB_TE_X, it turns out, within days of when the current eruptive phase of Kilauea began, in early 1983. And a volcano aptly describes my work on BIB_TE_X — a burst of activity, followed by a dormant stretch. (2004 will be more active than dormant.)

³ — to a Bell Labs Computer Science Colloquium audience that included some troff true believers

But back to 1983. Over the course of the next year and a half I implemented Lamport’s basic design, with a few enhancements. The first working version of BIB_TE_X (0.41) trudged forth in the summer of 1984. Lamport wrote, and Howard Trickey modified, a bibliography style based on Mary-Claire van Leunen’s suggestions in her *Handbook for Scholars* [9]. Trickey’s modified version was to become `btxbst.doc`, the template for BIB_TE_X’s four standard styles `plain`, `abbrv`, `alpha`, and `unsrt`. (By the way, these are called “standard” styles not because they are supposed to be some sort of standard, but because they are in the standard release of BIB_TE_X.)

The first public release of BIB_TE_X, in March 1985, was version 0.98, for L^AT_EX version 2.08. Several upgrades, including one for L^AT_EX 2.09, followed later that year. Version 0.99, which added many new features, was released in January 1988; two minor upgrades followed the next month, but BIB_TE_X itself has remained unchanged since then. The standard styles have been unchanged since March 1988.

In 1990 Karl Berry wrote some macros, for use in his `eplain.tex` package, that made BIB_TE_X usable with `plain` (and other versions of) T_EX. He and I modified the macros and released them as `btxmac.tex` in August 1990, usable with or without the `eplain` package. Several upgrades followed, the last in March 1995.

The current versions are: 0.99c for BIB_TE_X itself; 0.99b for `btxbst.doc` (the standard styles’ template file — but version 0.99a for each of the four standard styles); and 0.99k for `btxmac.tex`.

That’s for the software I’ve worked on. There has, in addition, been work that others have done to support BIB_TE_X, especially in the last decade:

- software for the handling of database (`.bib`) files (my personal database-handling preference is the text editor Emacs’s BIB_TE_X mode, but I don’t recommend that others join the Emacs religion just for that);
- the amassing of many database files for public consumption, minimizing the amount of work it takes to create new entries (Nelson Beebe has been particularly prolific in that regard);
- software for generating new bibliography styles automatically, without having to know details of the somewhat arcane `.bst` bibliography-style language (Patrick Daly’s `custom-bib/makebst` package has gotten widespread use — more on this later).

That brings us up to date.

Goals

BIB_TE_X has been very stable for some time now. Software stability is nice; it helps others build tools that augment the software. As suggested above, many tools have grown up around BIB_TE_X. The popularity of (L^A)_TE_X, however, has taken BIB_TE_X into places not anticipated, necessitating changes. Here are some general goals for BIB_TE_X 1.0:

- Easier nonstandard styles: The most frequent requests I see are for new bibliography styles. Creating a new bibliography style generally entails programming in the `.bst` language, which, for most users, is not an easy task. Patrick Daly's `custom-bib/makebst` package has made it reasonably easy for ordinary user to create new bibliography styles; for BIB_TE_X 1.0 the situation will improve even more.
- More international: BIB_TE_X has spread to the non-English-speaking world. BIB_TE_X 1.0 will address associated issues.
- More fields of study: The original BIB_TE_X users were from the technical world of mathematics and computer science. The BIB_TE_X 1.0 standard styles will do more to accommodate those in other fields, like the humanities.
- Enhanced sharing capabilities: There now exist huge `.bib`-file bibliographic databases, some available to users world wide. BIB_TE_X 1.0 will make the sharing of those databases easier.
- Better documentation: The BIB_TE_X 1.0 documentation will be more extensive.
- Froz_TE_N: For even better stability, BIB_TE_X 1.0 will be frozen. As with _TE_X 3.0, it will be upgraded for bug fixes only.

Some of the features planned for implementing those goals appear in the next section.

New features

Over the years I have accumulated a list of new features and probable changes for BIB_TE_X 1.0 and its standard styles. The list below is certainly not exhaustive, but it contains the most important items. Each one listed has a high probability of existing in BIB_TE_X 1.0.

- Software for generating customized bibliography style (`.bst`) files: In Santa Barbara I had claimed that there would be a Bibsty program to do that, similar in spirit to the processing that creates the four standard styles from the current template file `btxbst.doc`, but with lots more options. That Bibsty program, however, would partly duplicate what Patrick Daly has

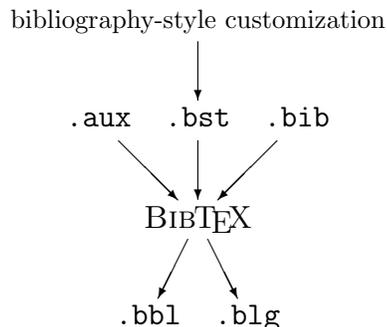


Figure 2: BIB_TE_X 1.0 input and output files.

done with the `custom-bib/makebst` package. So he and I have decided to collaborate for BIB_TE_X 1.0 on a system that will include the options of his current code, along with some things planned for the Bibsty program. This system will have, among other options, an easily changed symbolic name for each string that a bibliography style outputs to the `.bbl` file directly (such as ‘editor’ or ‘volume’), making it easier to, for example, convert bibliography styles from one language to another. Figure 2 shows how the new style customization system will fit into the scheme.

- Eight-bit input: Most current BIB_TE_X implementations can handle 8-bit input; BIB_TE_X 1.0 will guarantee, at a minimum, to support the character-set conventions of _TE_X 3. In addition, BIB_TE_X 1.0 may be able to accommodate Unicode.
- Multiple bibliographies within a single document: Many large documents contain several bibliographies—one bibliography per chapter for a book, or one per paper for a conference proceedings. Several solutions have arisen for handling such situations, but BIB_TE_X 1.0 will support multiple bibliographies directly.
- Enhanced sorting: The current BIB_TE_X does well with English, but doesn’t gracefully handle certain unusual sorting conventions of other languages, like Hungarian; BIB_TE_X 1.0 will have more powerful sorting capabilities for handling them.
- Reference-list back pointers: BIB_TE_X 1.0 will provide a direct mechanism for indicating in a reference-list entry where in the text that entry was cited. This is a useful feature that I suspect will become widespread now that our new typesetting technology makes it painless.

- An `@ALIAS` command: Suppose you have an entry in a database file that uses a different database key, say `knuth:tex`, from the cite key you prefer, say `texbook`; this might happen, for example, if that entry is in a shared database file. With BIBTEX 1.0 you will be able to keep the cite key and database key as is, as long as you put a command like

```
@ALIAS{texbook = knuth:tex}
```

somewhere in your database.

- A `@MODIFY` command: With BIBTEX 1.0 you will be able to create a slightly altered version of an entry. For example, you might want to add a `note` field to an entry in a public database file without having to repeat in your own personal database file all the information in that entry:

```
@MODIFY{laws-of-the-game,
  note = "Law~8 discusses restarts",
}
```

The `@MODIFY` command's database key should match the one from the public database.

- Distinguishing among identical database keys: If you are using two different database files that happen to use the same database key for different entries, you will be able to specify which entry you want by using a citation of the form

```
\cite{filename::database-key}
```

- A `.bib`-file extraction mode: BIBTEX 1.0 will have a mode that will let you extract just the information you need into a small `.bib` file. For example if you are submitting a paper to a journal that wants a `.bib` file in addition to a `.tex` file, but the bibliographic database you are using for the paper is huge, you can use the extraction mode to put just the entries you need for the paper into a separate `.bib` file that you can then send to the journal.
- A `\bibteoptions` (\LaTeX) command: Communication from (\LaTeX) to BIBTEX 1.0 will improve with this command.
- Extensions to the `\cite` command: Many citation styles aren't handled very gracefully by (\LaTeX)'s current `\cite` command. BIBTEX 1.0 and (\LaTeX) will more directly support more flexible `\cite` commands.
- Also appeared as: Sometimes a work appears in two different forms, for example as a journal article and then later in the author's collected works. BIBTEX 1.0 will have a mechanism to handle this.

- Name handling: BIBTEX 1.0 will have enhanced (but upward compatible) name processing, for example for complicated Spanish surnames, and for inverted-order Asian names.
- Standard-style changes: There will be lots of additions to the BIBTEX 1.0 standard styles, allowing many more options.
- New fields: The standard styles will also include a bunch of fields that are not now a part of the standard styles, including `day`, `isbn`, `issn`, `keywords`, `eprint`, `translator`, and `url`.
- New entry types: There will also be a few new entry types, including `@PERIODICAL` and perhaps `@PATENT`.
- `.bst`-language changes: There will be a few minor (but compatible) changes to the `.bst` language.
- `btmac.tex` update: These macros will be updated so that the user interfaces to BIBTEX 1.0 from \LaTeX and plain \TeX are as similar as possible.
- Documentation: The "BIBTEXing" and "Designing BIBTEX Styles" documents [5, 6] currently distributed with BIBTEX are neither as widely known nor as complete as they could be. For BIBTEX 1.0, the main documentation will be in a book, and will be much more thorough. Among other things, it will include a `.bib`-file grammar, so that those who are writing tools to manipulate the database files can make their software more robust.

Planned releases

2004 will see the release of an updated version of the `btmac.tex` macros. Probably toward the end of the year will come a bug-fix version of BIBTEX 0.99. (There are just a handful of actual bugs to BIBTEX itself, the most annoying being its mishandling of URLs, which didn't even exist when the current BIBTEX was released.)

There will be several beta releases of BIBTEX and its standard styles, leading up to 1.0; eventually one of those beta releases will be declared to be 1.0 and frozen.

Until BIBTEX 1.0 is finished, I will skim the `comp.text.tex` newsgroup for BIBTEX-related postings, so it suffices to post there anything you think I should see. In general that's a good place to post questions about BIBTEX—usually someone posts a useful reply. (Occasionally I'll reply by private email.)

References

- [1] Karl Berry and Oren Patashnik. `btxmac.tex`. Macros to make $\text{BIB}\text{T}\text{E}\text{X}$ work with plain TEX ; current version 0.99k, 13 November 1995.
- [2] Michel Goossens and Frank Mittelbach, with Johannes Braams, David Carlisle, and Chris Rowley. *The $\text{L}\text{A}\text{T}\text{E}\text{X}$ Companion*. Addison-Wesley, 2nd edition, 2004.
- [3] Helmut Kopka and Patrick W. Daly. *Guide to $\text{L}\text{A}\text{T}\text{E}\text{X}$* . Addison-Wesley, fourth edition, 2003.
- [4] Leslie Lamport. *$\text{L}\text{A}\text{T}\text{E}\text{X}$: A Document Preparation System*. Addison-Wesley, second edition, 1994.
- [5] Oren Patashnik. $\text{BIB}\text{T}\text{E}\text{X}$ ing. General documentation for $\text{BIB}\text{T}\text{E}\text{X}$ users, contained in the file `btxdoc.tex`, 8 February 1988.
- [6] Oren Patashnik. Designing $\text{BIB}\text{T}\text{E}\text{X}$ styles. Documentation for $\text{BIB}\text{T}\text{E}\text{X}$ style designers, contained in the file `btxhak.tex`, 8 February 1988.
- [7] Oren Patashnik. $\text{BIB}\text{T}\text{E}\text{X}$ 1.0. *TUGboat*, 15:269–273, 1994.
- [8] Unilogic, Ltd., Pittsburgh. *Scribe Document Production System User Manual*, fourth edition, April 1984. Chapter 12 and appendices E8 through E10 deal with bibliographies.
- [9] Mary-Claire van Leunen. *A Handbook for Scholars*. Oxford University Press, revised edition, 1992.

