BIBTEX Reconsidered

Dr. Reinhard Wonneberger
Electronic Data Systems (Deutschland) GmbH
Eisenstraße 56 (N15)
D-6090 Rüsselsheim
Federal Republic of Germany
Bitnet: qzdmgn@ruipc1e

Frank Mittelbach
Electronic Data Systems (Deutschland) GmbH
Eisenstraße 56 (N15)
D-6090 Rüsselsheim
Federal Republic of Germany
Bitnet: pzf5hz@ruipc1e

Abstract

The article contains a general discussion of BIBTEX, the proposals discussed at the EuroTEX89 meeting in Karlsruhe, and further proposals for extensions and changes to BIBTEX 0.99c.

Contents

1 Introduction

The more authors are won by TEX's outstanding quality of typesetting, the more it becomes obvious that typesetting is only one, though the core, step in producing a document.

What is needed in addition may by summarized by the term *document support.*[1]

One important part of document support is the concept of *General Markup* as realized in LATEX, DCF *Generalized Markup Language* [21], and, on the higher level of a *Document Definition Language*, by SGML [13, 7].

But two standard tasks of Document Support are not included in LATEX. One is indexing [9], the other one is producing bibliographies. The latter is handled by BIBTEX, which was developed by OREN

[1] On the following, see the discussion and practical examples in [42][181ff] and [4, 45].

Patashnik in close cooperation with Leslie Lamport. BibTeX was designed to work in close connection with LaTeX, but can also be used with other tools or as a stand alone program.[2] A comparison with bibliographic tools of other platforms can be found in [33].

For a long time, BibTeX was a niche product for experts only. This seems to be changing now:

1. A BibTeX requirements session was held at the EuroTeX meeting in Karlsruhe [6], resulting in a list of requirements that was spread out in [44] and published in [1, 24]. It is one of the purposes of the present article to elaborate these requirements.
2. BibTeX was discussed recently in TeXline [36, 34, 14]. The TeXline newsletter also began to print bibliographic information in BibTeX source format.
3. At our site at EDS, we introduced BibTeX about one year ago to help us with the large amount of inhouse and vendor documentation, which has to be handled in our industrial environment [47].

Thus it is one of the intentions of this article to help remove some of the obstacles which may prevent others from realizing BibTeX's benefits.

Although we will speak about alternatives and make quite a lot of suggestions, we should point out right away that we do appreciate what has been achieved so far. BibTeX is an excellent tool when used for the task it was designed for. Most of the problems arise because there are other needs that were not considered in the original design. The following lines are not more than a first try to present a collection of what might be desirable. Most of them adopt a user's, not an implementor's, point of view.

We are going to begin with a discussion of alternative concepts of citing. Next, we treat the links to LaTeX, and finally we deal with some issues confined to BibTeX itself.

2 Alternative approaches to citing

In many scientific areas, the citing information given in the text is kept to a minimum, e.g. a reference number. In some areas, however, different approaches to citing are required, and we shall consider a few of them here.

2.1 Presentational citing. The standard citing concept of BibTeX is based on the assumption that references are mainly used in rare circumstances for verification, but not read in the same way as the normal text.

There are, however, quite a lot of applications where bibliographic information is meant to be *read*, either alone or together with normal text. Such cases of *presentational citing* also need a different technical approach.

To get a vivid impression of why presentational citing might be desirable, the gentle reader of this paper might have a close look at footnote 3 on page 124.

2.1.1 Footnote citing. In some cases, authors may want their readers to see immediately those bibliographic items they are referring to.

Assembling such references in the back of the book or article will not do; it only makes sense if they are used rarely, so this case is quite similar to the alternative of footnotes vs. endnotes.

A good way to make bibliographic information easily accessible is to include it into footnotes; for an example, see [42]. This is discussed in more detail in Proposal 10 'Citefull style' on page 127.

2.1.2 Commented bibliographies. A similar question arises with commented bibliographies, where the commenting descriptions are to be mixed with the bibliographic information and *section* commands are to be used.

2.1.3 Publication lists. To document their own publications (*publish or perish!*), all scholars need publication lists which have some technical peculiarities.

Proposal 1 (Publication list)
Implement a `pubslist` *bibliographic style, with the following features: – omission or abbreviation of author's name, – chronological sequence, – entry related display of reviews (cf. Proposal 28 '*`@review` *entry' on page 130).*

2.1.4 Multiple bibliographies. In some cases, especially with textbooks for teaching, one would like to have bibliographies chapterwise. For a more technical discussion of this issue, see Proposal 14 'Selected bibliography generation' on page 128.

2.2 Short title citing. It may help to understand some of BibTeX's design limitations to have a look at another system, which was used to produce the bibliographies for some books of Reinhard Wonneberger.[3] In this system, a bibliographic entry looks like this:

```
:b t=l .label
:b     t=a .Author / Editor
```

[2] The first release (up to 0.98) is described under the roof of LaTeX [25, 39]. Substantial enhancements were introduced with release 0.99 [32]. The syntax for the bibliographic entries is upward compatible, but the style files have to be redone [31].

[3] [49] (explained in [42]), the first edition of [46] and the German edition thereof [41], and also [40] (which was typeset by the author's own Scriptor program), and [48].

```
:b     t=t .Title
:b         .(continuation line, for any type)
:b     t=u .Subtitle
:b     t=r .Series / Journal
:b     t=o .Address
:b     t=p .Publisher
:b     t=j .Year (edition)
:b     t=v .>out of<
:b     t=s .internal information
:b     t=e .(end mark)
```

The `:b .` is a GML-tag [21]; it carries a type attribute to specify the bibliographic field given in the remainder of the line. The '`v. >out of<`' field is used to refer to separate entries describing parts of a book, e.g. articles of a *Festschrift* (see below).

It is quite obvious that this system is far simpler than BIBTEX; it does not include adaptations along the lines of `.bst` bibliographic styles. But, in addition to some technical features worth noting, it may show some different attitudes and requirements when writing in fields like theology, literature, linguistics and the like.

1. With the few fields mentioned above, it was possible to accomodate more than 2,000 entries of very different kind. Typing is quite easy, and the simple input structure helps to get a quick and safe orientation in the source files, cf. Proposal 12 'Structured source layout' on page 128.
2. The `r`-field is ambiguous, being used for journals, e.g. `ZAW 92 (1980) 185-204`, in which case the year is given here, and for series, e.g. `BZAW 123`, in which case a separate year field is used. Though it would need some analysis to extract the year of an article, this scheme is very efficient for input.
3. The most interesting aspect is perhaps the inclusion of *incollection* type items. The need arises because books and the parts of them, e.g. articles, always get different entries. Let us have an abbreviated example, the first entry giving the collected volume, the second one giving the article contained in it:

```
t=l .Weydt Partikeln
    t=a .Weydt, Harald (ed.)
    t=t .Die Partikeln ...    [book]
    t=j .1979
    t=v .>Wonneberger Pragmatik<
t=l .Wonneberger Pragmatik
    t=a .Wonneberger, R.
    t=t .Zur Syntax ...    [article]
    t=r .Weydt Partikeln 488-499
```

The interesting point is that the bibliography (corresponding to the `.bbl` file) is produced in *one pass.*[4]

4. There is also a difference in the citation mechanism. References to literature are done as index entries, which are also printed in the text; to give an example, the text entry `:s.12 Wonneberger Normaltext 99`, which will appear in print as '*Wonneberger Normaltext* 99', also produces an index entry. This gives an index of quoted literature, which is quite useful in the writing of bigger manuscripts. A file with bibliographic requests is produced during index processing. As entries are sorted anyway for the index, no special sort is needed to get the bibliographic request file into alphabetic order. All special symbols like *umlauts* are replaced by a transcription like 'ae', 'ss' and the like during label processing.

The last point seems to be of general importance. As an author, I do not want to be dependent on several LATEX and BIBTEX runs to get my cite keys resolved. I want to code an explicit short title in my source, the meaning of which is immediately clear to myself and to others working in this field, and I also want to identify who was cited where.

Proposal 2 (Explicit labels)
Allow explicit labels, like "`Wonneberger Normaltext`*", which can be printed directly and used with* `\nocite`. *Umlauts can be normalised.*

Remarks
A similar system is used in linguistics, giving author *and* year, *e.g.* Chomsky (1957c). *This case seems to be more complicated, because punctuation may be subject to stylistic guidelines, and counting relative to the year is context dependent.*

It is important to understand that these different approaches to citing are by no means just a matter of taste. In most cases they reflect a specific functionality and perhaps also general attitudes of a community of scholars.

Speaking in terms of Computer Science, the familiar BIBTEX method is *addressed* citing, whereas these methods can be seen as *associative* citing, which means: addressing by content.

Addressing does not matter as long as it is kept inside the machine. It becomes a handicap if the user is involved.

The question of associative citing also evokes the topic of databases (see below).

[4] When the list of bibliographic requests (corresponding to `\citation`) is matched with the bibliographic database, all items are checked for a possible 'v' field. Then the program goes the other way round and checks whether the label framed by angle brackets is requested in the bibliographic requests list. It is not checked, however, whether the `r`-field matches with the book label.

2.3 BibTeX in the industrial field. At first sight BibTeX might seem to be an academic tool. This is true because of the academic need for quoting others.

But an important benefit of BibTeX can only be realized by sharing bibliographic information, and sharing resources might be more common in an industrial environment.[5]

There are some requirements which result from industry needs. Internal labels will not normally be built on authors' names, as many publications do not show their authors at all. In these cases it may be best to build on the publication number, which, as a rule, is also printed on the publication itself.[6]

In an MVS environment one might wish to have only one document identifier which can serve both as a label and as a member name for source and object forms of the document. This, however, would impose the restrictions for member names on the label.[7]

Proposal 3 (Worldwide convention(s) for labels)
Agree on a standard for the formation of cite-keys.

In an environment with demand printing, some publications will be available for general printing, e.g. [47] is made available as [43]. Instead of using two separate entries, it might be nice to have fields that can be switched on and off dynamically. This concept is also important for maintaining additional information within one entry which should be used only sometimes, e.g. prices, etc. While it is possible in principle to write different style files that support different fields, this overhead is error prone.

Proposal 4 (Switchable fields)
Support the inclusion of individual fields on request.

Remarks
This can be achieved with the help of the `@like` *option (cf. Proposal 26 'Like option' on page 130) since unwanted fields can then be identified with ignored fields. Alternatively, it could be supported in a substyle concept (cf. Proposal 22 'Substyle concept' on page 129).*

3 Interaction with LaTeX

3.1 Multiple bibliographies. BibTeX was designed for documents containing one reference list, i.e. one `thebibliography` environment. Therefore it was sensible to read the citation information directly out of LaTeX's `.aux` file. If more than one bibliography is necessary, one can circumvent the problem by using the sub-level `.aux` files provided by the `\include` command and reading the resulting `.bbl` files manually.[8]

Alternatively the `\bibliography` command can be changed slightly to read in a `.bbl` file corresponding to the current include file. This will give us one bibliography per `\include` command, a command that will always start a new page. Therefore this mechanism is not adequate for journals in which the articles are printed just one after the other. It is also not usable for manuals, etc., that have several reference lists, one after the other, e.g. 'Further reading', 'Related publications', etc.

A second alternative is to change the `\cite` command to write the citations not into the `.aux` file but into other files, one for each wanted reference list. This will allow one to use the resulting BibTeX output at any place, but will require a considerable number of redefinitions of LaTeX commands. Furthermore, both methods are very time consuming and error prone because BibTeX has to be started for every file generated.

Proposal 5 (Multiple bibliographies (LaTeX))
There are plans to extend the bibliography support in the re-implementation of LaTeX in such a way that several bibliographies within one document (all generated by BibTeX*) are possible. The syntax for this feature is currently under discussion.*

Remarks
This support is only possible in an efficient way if the BibTeX *input syntax is changed according to Proposal 14 'Selected bibliography generation' on page 128.*

3.2 Title formatting. To use BibTeX for the formatting of a title page could be another useful and efficient application. It would be an advantage

[5] See the corresponding considerations for TeX in [47].

[6] The IBM system of form numbers shows how a very big publisher handles this [20]:

`ABBB-CCCC-EE`	Typical IBM Documentation Number
`A...-....-..`	Use Key, e.g. `G`: generally available, `S`: for sale; `L`: for licensees
`.BBB-CCCC-..`	Form Number
`.BBB-....-..`	Prefix, e.g. one number for all DCF publications; specific prefixes denote publication categories like Logic Manuals, Bill of Forms, Microfiche, Pseudonumber, Technical Newsletter, Supplement; for details see [20].
`....-CCCC-..`	Base Number relative to the prefix
`....-....-EE`	Suffix, e.g. 00 for first edition

The first three elements might be used right away as a BibTeX label, assuming that new editions should be quoted automatically.

[7] Max. 8 alphanumerics, starting with a letter (including the national characters `@, #, $`).

[8] The new LaTeX will use only one `.aux` file, so that this mechanism will not work any longer. For further information see [27].

that arbitrary information could be gathered in the BIBTEX database while each journal provides its own title-.bst that picks up the information of interest. This style would then produce an output file containing LATEX commands for formatting a title according to the house style. An additional benefit would be that each paper produced in this way would already contain its own bibliography entry in BIBTEX format.[9]

Proposal 6 (BIBTEX and title page)
Evaluate whether such a scheme is feasible and easy to use. If so, devise new LATEX commands that handle this concept in a user friendly way.

Remarks
This clearly requires multiple bibliographies (cf. Proposal 14 'Selected bibliography generation' on page 128), as one cannot ask the user for an additional BIBTEX *run to produce a title page. Such a feature should be clearly considered as a suggested add-on, but a conventional way to produce a title in a document should still be available.*

BIBTEX can be particularly helpful in providing title information for software in a complete and standardized way.

Proposal 7 (BIBTEX software titling)
Develop a concept for self-contained software title information along the proposal in [2][486–487].

3.3 Formatting commands. With the release of BIBTEX 0.99 it is possible to include LATEX commands in the bibliography database that will be copied by BIBTEX to the very beginning of the .bbl file. This allows one to specify formatting declarations and to introduce new commands which will be used in the bibliography entry. But LATEX supports only \newcommand and \renewcommand. If for example the database contains the line

```
@preamble(\newcommand{\WEB}{{\sc web}})
```

it is not possible to use this database together with the ltugboat style option because the command \WEB is already defined in this style and the \newcommand will therefore balk.

Proposal 8 (Supply command)
Add a \supplycommand[10] *macro to LATEX which behaves like* \newcommand *if the command name is previously undefined but does nothing (or prints only a warning) when the command already has a meaning.*

3.4 Decentralized bibliography. Some document types require bibliographic entries to be placed directly into the text or into a footnote, cf. section 2.1 on page 124. To support this, BIBTEX should be changed to allow sequential reading of the .bbl file.

Proposal 9 (Singularize .bbl)
Change the output of BIBTEX *by surrounding the whole entry with a pair of braces, i.e.*

```
\bibitem{Kn86}{%
  Donald E. Knuth. \newblock
  ...
}
```

Remarks
Using the \read *command to access the* .bbl *information would forbid commands that change* \catcode *(like* \verb*) in the database.*

3.5 Sequential citations. As mentioned in section 2.1 on page 124, a form of sequential citing, e.g. in footnotes, is a common requirement in certain disciplines. This should be supported by LATEX at least via a style option.

Proposal 10 (Citefull style)
Write a LATEX style option that supports this sort of citation. This would necessarily include a new \citefull[11] *command that marks the place where the full entry should be cited. The* \cite *would then refer to this place.*

Remarks
This requires a different syntax in the .bbl *file as explained in Proposal 9 'Singularize* .bbl*' on page 127.*

3.6 Internal names. After looking into an .aux file many users get confused by the fact that user commands (e.g. \cite, \bibliographystyle, etc.) have nearly identical counterparts in internal commands to be understood by BIBTEX. Unfortunately these commands (e.g. \citation, \bibstyle, etc.) are user accessible, which may lead to confusion.

Proposal 11 (Change internal names)
Change the internal names that are written by LATEX into the .aux *file, so that they cannot be accessed by the ordinary user.*

4 BIBTEX input

4.1 Source organization.

[9] This is similar to the practice to print a short bibliographic entry as provided by the national library on the backside of the title page.

[10] The name is only a suggestion. A simple definition for macros without parameters could be:

```
\def\supplycommand #1#2{\ifx
  \undefined #1\def #1{#2}\fi}
```

[11] The name is chosen to apear near \cite in the alphabetical sequence and to allow incremental search in appropriate editors.

4.1.1 Uniform source layout. BibTeX allows identical information to be coded in different syntaxes. For large bibliographies, it is important to find a standard format of coding, cf. the rather uniform source layout of the Reduce bibliography [12].

Proposal 12 (Structured source layout)
Develop a convention to structure BibTeX *source files for better readability.*

Remarks
This also implies agreed positions for commas and a convention about braces vs. brackets vs. quotes.

4.1.2 Error recovery. "In practice, we all make mistakes. And one of the most common typographic errors is to forget a '}', ...". These words of Knuth [23][205] about TeX also apply to BibTeX. If braces or quotes do not match properly, it can be quite difficult to detect the corresponding location. In some cases, the error will be reported only at the end of the file, and many entries may have been skipped.[12]

Proposal 13 (Entry separator)
Define an empty line to act as an entry separator.

Remarks
This strategy is similar to the one adopted for `\par` *in TeX [23][205]. It would only prevent empty lines from occuring inside entries, which does not seem to be a big loss.*

4.2 Supporting multiple bibliographies. As explained in section 3.1 on page 126 one of the important shortcomings of BibTeX is the missing support of several bibliographies within one document. This could be incorporated by changing the syntax of internal commands in the following way:

Proposal 14 (Selected bibliography generation)
Change the commands recognized by BibTeX *in the* `.aux` *file so that they accept two arguments with the following meaning:*

- *The first argument is a list of context selectors that will help* BibTeX *to decide which citations to choose for a particular* `thebibliography` *environment, which data bases to search, and which style to use for the bibliography to generate.*
- *The second argument holds the cite key, the style, or a list of databases, respectively.*
- *A context selector is a unique string of at most eight characters.*[13]

In detail, the internal commands found in the `.aux` *file should be interpreted in the following way:*

`\bibstyle` *If the first argument is empty, generate a bibliography containing all citations searching all databases. Otherwise, the first argument contains a list of context selectors. In this case, generate for every context selector a bibliography considering all citations and databases that have been specified for this selector.*

More than one `\bibstyle` *command in the* `.aux` *file is allowed. But every context selector should be used only once.*

`\bibdata` *If the first argument is empty, the second specifies a list of data bases which should be used to search for* all *citations. Otherwise, the data bases from the second argument are searched only if the current citation contains at least one of the context selectors.*

Again, more than one `\bibdata` *command is allowed. Therefore, the problem of a buffer overflow (when writing to the* `.aux` *file) can be prevented.*

`\citation` *If the first argument is empty, the citation will be included in every bibliography generated by the* `\bibstyle` *commands in the* `.aux` *file. If it contains a list of context selectors the citation will show up in every bibliography generated for them.*

Remarks
The internal command names should be changed according to Proposal 11 'Change internal names' on page 127.

It seems advisable to process different `\bibstyle` *commands sequentially to avoid the problem of memory overflow.*

4.3 Language considerations. A general presentation of language-related problems can be found in [17].

4.3.1 Language flags. With TeX 3.0 it is possible to support several languages within one document. This is important for hyphenation, use of fixed names (like "volume", "and", etc.) but also for direct entry formatting. There is, for example, no decapitalization of titles in German.

Proposal 15 (Language flag)
Add a language flag for entry processing by both BibTeX *and TeX.*

Remarks
This should be available for every field, but the possibility to specify a language for the whole entry (cf.

[12] For the time being, such errors may be detected by changing all entry escape symbols '`@`' into something like '`",})@`'. These closing delimiters will trigger error messages if the matching is wrong; otherwise, they will be ignored.

[13] This restriction seems to be necessary to allow a standard naming scheme for the resulting output files. Of course, the possibility to construct the output file names from the TeX job name (e.g. `\jobname.bbl`) has to be given up on most systems.

Proposal 34 'Language field' on page 131) is equally important.

Until this has been achieved, standard styles should be language independent, e.g. *Computer Physics Communications 61* instead of *volume 61 of Computer Physics Communications.*

Proposal 16 (Language independent standard styles)
The BibTeX *standard styles should be language independent.*

4.3.2 Symbols. Special symbols like *umlauts* play an important role in an international environment.

Proposal 17 (Symbol convention)
Though BibTeX *makes provisions for some special cases, it should also be able to handle language related symbols (e.g. those from* german.sty *[30, 29]) and allow outputting them in a different syntax.*

To produce output for other typesetting systems, e.g. for DCF, BibTeX should be able to output language related symbols in a different syntax. To give an example, the umlaut `{\"a}` should be converted to `&a.` for DCF.

Proposal 18 (Symbol conversion)
Set up symbol conversion with the help of an additional style file at run time similar to the way MakeIndex *handles different syntax conventions for input and output [9].*

Proposal 19 (Composed symbols)
Composed symbols should be generated. Roman figures, e.g. should be kept as arabic numbers, with some kind of a transformational prefix like in DCF, *where* `&R'4.` *will give 'IV', whereas* `&r'6.` *will give 'vi'.*[14]

4.3.3 Sorting. Different character sets require different sorting rules [22]. In the current BibTeX, the use of accented characters in author names or titles often leads to incorrect bibliographic sequences in the output.

Proposal 20 (Sorting rules)
Devise some mechanism to incorporate customizable sorting rules into BibTeX.

Remarks
The same module should be used for MakeIndex, *where similar problems arise.*

4.3.4 Names. The handling of names in BibTeX is very sophisticated. Nevertheless it cannot handle all classes well enough for general applications. For example, titles in the name fields are misplaced in certain styles,[15] and monks and nuns should be allowed to have their congregation indicated, e.g. 'James Swetnam, S.J.'.

Proposal 21 (Names and titles)
Reconsider the BibTeX *name function and extend it in such a way that it can deal with foreign names.*

Under the present scheme, agree to write all names with a comma, so that also Spanish names will be unambiguous, e.g. 'García Márquez, Gabriel'.

Remarks
Alternatively, one could give the style file more access to reprogram the function, but this would not solve the problems of an international data base.

4.4 Style considerations.

4.4.1 Substyles. At the moment a BibTeX style is responsible for both the logical structure of the input (e.g. which fields and entries are recognized) and the structure of the output (e.g. sorting, label generation, formatting of the entries, etc.). Since these tasks are, at least in principle, unrelated to each other, it would be better to separate them.

Proposal 22 (Substyle concept)
Determine a substyle concept for bibliography styles that allow to change certain aspects of the formatting and input recognition similar to the `\documentstyle` *command of LaTeX. This concept could be used to handle the requirements discussed in Proposal 20 'Sorting rules' on page 129, Proposal 34 'Language field' on page 131, and Proposal 26 'Like option' on page 130.*

4.4.2 Special styles. When writing documents it is helpful to have a printed version of the BibTeX database which shows the names of the citation keys.

Proposal 23 (Cite style)
Add a substyle to BibTeX*'s standard distribution that additionally shows the citation keys.*

Remarks
Such a style could be derived directly from the standard styles. Using a suitable preprocessor (see Pro-

[14] The following macros could be used in a LaTeX style file to interpret such input:

```
\def\R'#1.{\@Roman{#1}}
\def\r'#1.{\@roman{#1}}
```

[15] Dr. is part of the name in Germany, e.g., so one should get 'Dr. R. Wonneberger', but 'Dr. Wonneberger, Reinhard' or 'Wonneberger, Reinhard, Dr.', if the title is not dropped at all, as it is customary in some academic fields. Curious as it may sound, the title was added in the second edition of [39] without the the author's knowledge.

posal 25 'Style distribution' on page 130), it should be no problem to provide it. An example style was given in [11]. Alternatively, this can also be achieved with a suitable style option in LaTeX.

When submitting a paper like this one for publication, the bibliography will have to be sent in `.bbl` format. It is very difficult then for the editor to make modifications to the bibliographic style, in contrast to the fact that they can be done quite easily with a LaTeX `\bibliographystyle` command.

Proposal 24 (Collect style)
Add a substyle to BibTeX's standard distribution that will collect the subset of bibliographic entries required for the document and write it into a separate `.bib` *file.*

Remarks
Such a style will also allow editors to produce a combined bibliography for the whole volume, provided that labels can be mapped as described in Proposal 38 'Multiple labels' on page 131.

4.4.3 Documentation and distribution. The main bibliography styles (`abbrv`, `alpha`, `plain` and `unsrt`) are derived from one master source (`btxbst.doc`) which is written using a C preprocessor syntax. The individual styles are then extracted by setting individual preprocessor flags. Unfortunately the choice of a C preprocessor seems to be tied to a UNIX environment. This means that installations without a C compiler must rely on supplied `.bst` files. The amount of time involved to customize a set of `.bst` files in such an environment is extremely high because every `.bst` file has to be changed individually.

Proposal 25 (Style distribution)
Supply and document all standard styles with a tool that is generally available in all TeX distributions. Since `WEB` *has the disadvantage of being tied to* PASCAL, *we propose using the extended version of* `doc.sty` *and* `docstrip.tex` *[26] which is capable of processing conditional code.*

4.5 Entries and fields. As shown by David Rhead [34], the names and meanings of entries and fields provided by the standard styles do not conform to the classification given by [8], [38], or [5]. But even if a more standardized set of entries and fields were implemented, one would certainly find exceptional applications of BibTeX that involve just another class of documents. The problem of different naming conventions is that it will be difficult to merge databases. This can be circumvented if BibTeX is able to interpret field and entry names in a database depending on exception rules.

Proposal 26 (Like option)
Add a `@like` *entry that might have the following syntax:*

```
@like(⟨entry1⟩)(⟨entry2⟩)(⟨linklist⟩)
```

This entry will tell BibTeX *that any ⟨entry1⟩ in the data base should be treated as an entry with name ⟨entry2⟩. This sort of command is useful only if both entries have similar fields. The ⟨linklist⟩ should be used to set up conversions between individual fields, e.g. the command*

```
@like(edsmanual)(manual)
(intnote= note
printnote= note)
```

tells BibTeX *that it should interpret an* `@edsmanual` *entry as a* `@manual` *entry and convert any* `intnote` *or* `printnote` *field into a* `note` *field. The ⟨linklist⟩ can be empty.*

Remarks
These additional commands can be placed in a separate database that is loaded first. The example above can have the result that after conversion more than one field of the same sort exists. This should be handled according to Proposal 33 'Field loops' on page 131.

4.5.1 New entries. There are many possible additions to the current set of standard entries. A proposal for a rather different set of entries is given in [34]. If the current set of entries is kept as a basis, we would suggest adding at least some new ones.

Proposal 27 (`@journal` entry)
At the moment there is no good way to cite an entire journal. This would be required for cross references but is also of interest to draw attention to general topics covered in a journal without citing specific articles.

Proposal 28 (`@review` entry)
A review will typically have to `crossref` *another bibliographic entry as part of its title.*

Review being a publication class of its own, it should be identifiable by its entry name.

Proposal 29 (`@product` entry)
Citing a product (e.g. program, hardware, etc.) is nearly impossible since no official entry contains suitable fields.

Proposal 30 (`@institution` and `@person` entry)
These entries should accomodate all kinds of addresses and allow `person` *to* `crossref` `institution`.

4.5.2 Fewer entries. As opposed to adding further entries, it might be better to use only a few fundamental ones like `monograph`, `periodical` etc., cf. [34], and give a more detailed classification in a `type` field.

Proposal 31 (Review entries)
Discuss entry concept.

4.5.3 Stacked entries. In some cases, the *incollection* relation involves more than two levels, e.g. this paper is part of a proceedings volume which in turn is part of a journal.

Proposal 32 (Stacked entries)
Allow stacked entries.

4.5.4 Repeated fields. In many cases it makes sense to have identical fields within one entry. Consider, for example, a citation giving the years for two editions like: "1.Aufl. 1980, 2.Aufl. 1982". It is also useful simply to combine certain information as explained in Proposal 26 'Like option' on page 130.

Proposal 33 (Field loops)
Extend the BibTEX *style language to handle multiple fields. If desired by the style designer, it should be possible to express that certain fields are bound together in the sense that the designer can access the first occurrence of a number of fields (e.g.* `year` *and* `edition` *in the above example), then the second and so forth.*

Remarks
If a field of an entry is not marked as multiple *in the sense above, all occurrences of this field should be simply concatenated with a programmable separator string.*

4.5.5 Supported fields. A language field should be available for the language of the publication.

Proposal 34 (Language field)
Add an optional field `language` *to each entry, specifying the language used within the entry. If no field is specified, a default language should be used.*

Remarks
The default language can be implemented as part of a substyle concept (cf. Proposal 22 'Substyle concept' on page 129). The language of individual fields (see Proposal 34 'Language field' on page 131) should always take precedence over the general or default language. This should not be confused with the case where the book is in a language different from the bibliographic entry, e.g. Hebrew and Russian publications when quoted with the bibliographic information in English. In this case the original language can be placed into the `note` *field.*

One important piece of information for published items is their *International Standard Book Number* (ISBN) or *International Standard Series Number* (ISSN), which are useful to identify and order publications.

Proposal 35 (Number field)
Support a field for ISBN and ISSN numbers or a general number field for all entries.

Remarks
It is possible that several numbers exist for one entry (e.g. a hard- and softcover version for [23], etc.); see Proposal 33 'Field loops' on page 131 for ideas on handling this situation.

The current use of `booktitle` is difficult to understand.

Proposal 36 (`booktitle`)
A better scheme should be devised for the `booktitle` *concept in connection with* `crossref`*. At least, booktitle should be made optional.*

4.5.6 Unsupported fields. In the current BibTEX implementation, there is one field, namely `annote`, which is not supported by the standard BibTEX styles. By defining such names BibTEX databases are kept portable.

Proposal 37 (Standardize extended field names)
Add names for all fields to the user documentation of BibTEX *which are of interest in extended styles. This would include fields like* `abstract`*, etc.*

4.5.7 Labels. As long as labels are not defined by the publisher, different labels will be used for one and the same element. This prevents an easy combination of databases.

Proposal 38 (Multiple labels)
Discuss multiple cite labels or alias-fields (to allow different systems of reference, e.g. in different user groups), or a scheme of citing by content [33].

Remarks
An application for this is given in Proposal 24 'Collect style' on page 130.

4.6 Crossreferencing. BibTEX's possibility to crossreference entries (introduced in version 0.99) is restricted to a fixed order of entries (referenced entries must follow later in the data base). This is necessary since BibTEX is essentially a one-pass system. On the other hand, this poses great problems in maintaining huge databases.

Proposal 39 (Multi-pass system)
Remove this restriction by reading the data base several times if necessary.

Remarks
A multi-pass system is also necessary to support Proposal 14 'Selected bibliography generation' on page 128 and perhaps also Proposal 22 'Substyle concept' on page 129.

Another possibility is to allow backward references from whole publications to parts. This would allow for processing everything correctly in one pass and help to keep track of consistency.

Proposal 40 (Backward references)
Consider this approach as an alternative to the current concept.

Remarks
This would also allow to identify cited parts of a collect volume under its entry.

The concept of crossreference might be extended to other areas of application.

Proposal 41 (Extended crossreferencing)
Allow more than one crossreference for an entry.

Remarks
This would allow to crossref persons and institutions, e.g. authors with their affiliation, or publishers with their address, in addition to the normal crossreferencing of collect volumes.

4.7 Compatibility with professional bibliographic databases. Further development should consider standards like RAK. RAK (*Regeln für die alphabetische Katalogisierung*) is the present standard for German scientific libraries (and a successor to the so-called *Preußische Instruktionen* (Prussian Instructions). In addition to giving rules on how to deal with publications without an explicit author or editor, this standard specifies the full range of fields that are necessary to describe all types of items held in such libraries.

Proposal 42 (RAK compatibility)
Though this standard is far too complicated for nonprofessionals, it might be useful to think about a way to inherit BIBTEX *fields from a professional catalog entry.*

Remarks
This topic could best be dealt with if we could find a professional librarian who is also familiar with TEX.

5 BIBTEX output

When working with larger bibliographies, it is important to have a working printout which also shows internal data. Lookup in a printout according to our experience is much more efficient than lookup in several files.[16]

5.1 'Comefrom'-information.

5.1.1 Date and time. For long-living documents it is often essential to determine when the document was last updated. On many systems, the automatic date and time stamp of a file is not an adequate source of information, because it might change during physical movement of files, etc.

Proposal 43 (Date and time)
Write the date and time of execution into the `.bbl` *and* `.blg` *files using a TEX acceptable format, i.e. starting lines with a percent sign or using appropriate macros.*

5.1.2 Source. In a large bibliography, it can be difficult to locate a specific entry. It is useful to have a working copy which, in addition to the cite label, also shows the file name for each entry.

Proposal 44 (Source information)
Add the source file information to each `\bibitem` *command on demand.*

5.2 Error messages. If braces or quotes do not match properly, BIBTEX may read bulks of lines in the vain hope of finding a good end delimiter (cf. Proposal 13 'Entry separator' on page 128).

Proposal 45 (Error message entry identification)
Identify the entry and field being processed in BIBTEX *error messages to help with spotting long range syntax errors of the input.*

Proposal 46 (Warning message for unknown fields)
Give a warning message for unknown fields, unless declared before, or switched off (cf. Proposal 37 'Standardize extended field names' on page 131).

5.3 Database maintenance. BIBTEX's databases are simple ASCII files. This has the advantage that such files are easily portable. On the other hand, maintaining these files is difficult. If, for example, two large databases are to be merged, unifying the citation keys is a time consuming task. Using the `\cite{*}` feature of BIBTEX can help a little bit in this regard, but if the database has too many entries it will blow up BIBTEX's memory, e.g. the Reduce bibliography [12] with about 40 pages is not processable on most installations.

Proposal 47 (Nonsorting mode)
Add a feature to BIBTEX *that allows sequential reading of arbitrarily large databases. The only information which should be saved is the citation key and*

[16] Similar considerations apply to LATEX, where a general `draft` substyle showing 'comefrom' and crossref information is highly desirable.

its position in the database so that multiple entries can be detected.

Remarks
This is not the same as writing a style file which ignores nearly all fields. The latter can be used only to detect multiple citation keys while the proposed feature would allow for printing all information contained in the database.

5.3.1 Comments. In a large database it is often useful to include comments (about the last update, etc.). While this is possible between entries, as long as one does not use one of BIBTEX's special characters, it is not allowed inside a bibliography entry. If, for example, not all information is available, it would be nice to comment out open fields instead of deleting them from the database. This is especially useful in an environment where the user is guided by templates which show all possible fields.

Proposal 48 (Comment character)
Make the percent character a BIBTEX *comment sign to allow comments in the database.*

6 BIBTEX internal commands

6.1 `.bst` Stack language documentation. The 'unnamed language' as described in [31] is difficult to understand even for experienced programmers since it implements the seldom used concept of a stack language. But the main clients of BIBTEX in the future are probably people with only a small or no knowledge in computer science. They will have to change certain aspects of the standard styles and for that reason have to understand the basic principles of the language.

Proposal 49 (Progammers guide)
Expand the BIBTEX *programmers guide in a way that even inexperienced people are able to customize existing* `.bst` *files.*

In any case it is questionable whether the concept of a stack language is natural for the given problem of producing bibliographies.

Proposal 50 (Style language change)
Consider the possibility of changing the style language of BIBTEX *in the future to a different concept.*

Remarks
If the necessity of customizing supplied style files is lessened by introducing a substyle concept (cf. Proposal 22 'Substyle concept' on page 129), one could perhaps leave the language as it is: as an amusing or frustrating experience for a few gurus.

6.2 The incompleteness of the language. When one of the authors (Frank Mittelbach) implemented a `tugproc.bst` according to the guidelines for TUG Proceedings [10], he ran into several problems. It was pretty difficult to convert explicit dashes (i.e., '`--`' and '`---`') into the commands `\dash` and `\Dash` that should be used instead. Problems arose because `text.prefix$` and `text.length$` operate on special characters. Another dilemma was produced by the requirement that punctuation characters should go into the quotes. As a result the `add.period$` could not be used any longer.

Proposal 51 (Extend internal commands)
Check whether standard requirements for bibliographies can be programmed in a straightforward way with the given set of commands; extend this set if necessary.

7 Implementation and organization

7.1 Sharing bibliographic information. It takes quite a lot of know-how, time, and care to produce a good entry for BIBTEX. Much can be gained by distributing precise bibliographic information, as it is shown by the Reduce bibliography [12].

Proposal 52 (`.bib` distribution)
Discussion lists like [35] and similar installments should encourage distribution of bibliographic information in `.bib` *format.*

Even without the sharing of source files, it is much easier to type a ready-made BIBTEX entry from paper (e.g. [36]) than to reconstruct it from normal bibliographic information.

Proposal 53 (`.bib` printout)
Following the good example of TEXline [36], also other publications like TUGboat [37] should print bibliographic information in `.bib` *format.*

7.2 Software maintenance with BIBTEX entries. The quality of software available to the TEX community varies from exceptional to poor. One of the main problems is that, quite often, it is not easy to determine author, release, last update, etc. for a macro package or a program source. This could be prevented if a standard for documentation is developed which describes the relevant facts. Here BIBTEX would be the ideal tool.

Proposal 54 (Software entries)
Agree on a small set of entries with corresponding fields that describe software. This standard should be used within the TEX community for all software distributed. Authors submitting software to servers

should be asked to add such an entry in front of their contribution. Preferably fields like `abstract` *should be included so that this information can find its way directly into a local guide.*

7.3 BibTeX status in the TeX community. With this paper we want to stress the importance of companion programs to TeX, BibTeX in this case.

Proposal 55 (Standard TeX support)
BibTeX, and MakeIndex[17] accordingly, should get the same support by the TeX Users Group (TUG) as TeX and LaTeX.

Some of our suggestions will be quite easy to implement even under the current concept of BibTeX, some others will need extensions, and still others may require a completely different implementation.

Proposal 56 (Academic research)
Doing further research into the questions raised in this paper might be a challenge for students and teachers who are interested in computer science as well as in librarianship. Such research might also foster cooperation among different disciplines and experienced practitioners.

Remarks
David Rhead, *known for his work on BibTeX [34], has offered to help with coordinating further work. Anyone who wishes to volunteer, or who is in a position to supervise a research student doing further work, is invited to send details of their interests to David.*[18]

The present authors would be pleased to supply their manuscript to serve as a nucleus for further work on BibTeX.

Proposal 57 (Living document)
Create a living document to display requests, results of discussion, and the status of implementation for BibTeX.

References

[1] Bailey, Rosemary. "TeX89 looks at LaTeX tools." *TeXline*, (10):8, 1990.

[2] Beebe, Nelson H.F. "From the president." *TUGboat*, 11(4):485 – 487, 1990.

[3] Beeton, Barbara. "Report from the question and answer session." In Durst, Lincoln, editor, *1990 Annual Meeting Proceedings*, pages 455 – 458, Providence, September 1990. TeX Users Group. TUGboat 11 (1990/3).

[4] Brandt, Josef. "Computer-aided production of scientific documents." Pages 163 – 176 in *Man-Machine Interface in the Scientific Environment. Proceedings of the 8th European Summer School on Computing Techniques in Physics. Skalský Dvůr, Czecholsovakia, 19 – 28 September 1989*, J. Nadrchal [28]. Invited Paper.

[5] British Standards Institution. *Citing publications by bibliographic references*, 1987. BS 5605.

[6] Brüggemann-Klein, Anne, editor. *Proceedings of the 4th European TeX Conference, September 11th – 13th, 1989*, Karlsruhe, forthcoming. EuroTUG.

[7] Bryan, Martin. *SGML: An Author's Guide to the Standard Generalized Markup Language.* Addison-Wesley, Woking, England; Reading Massachusetts, second edition, 1988.

[8] Butcher, Judith. *Copy-editing.* Cambridge University Press, 2 edition, 1981.

[9] Chen, Pehong. MakeIndex*: A General Purpose, Formatter-Independent Index Processor.* University of California, Berkeley, June 1988. LaTeX version of a file distributed with MakeIndex (MakeIndex adaptation manual); source in 'adeit.mkindex.tex(pehong)'.

[10] Durst, Lincoln K. "Guidelines for *proceedings* of the annual meeting of the TeX users group." Received February, 1990.

[11] Hailperin, Max. "Another response to BibTeX question." *TeXhax*, 90(40), 22 April 1990.

[12] Hearn, Anthony C. *REDUCE Bibliography.* The RAND Corporation, June 1990. Available as `.bib` source; enquiries to network address: reduce @ rand.org.

[13] Herwijnen, Eric van. *Practical SGML.* Kluwer, Dordrecht, NL, 1990.

[14] Higgins, Christopher P. "Cross referencing the bibliography." *TeXline*, (10):8 – 9, May 1990.

[15] IBM National Language Technical Center. *Designing Enabled Products, Rules and Guidelines.* Volume 1 of NLIDG [17], 1987. IBM Order Number: SE09-8001-00.

[16] IBM National Language Technical Center. *Left-to-Right Languages and Double-Byte Character Set Languages.* Volume 2 of NLIDG [17], 1987. IBM Order Number: SE09-8002-00.

[17] IBM National Language Technical Center, editor. *National Language Information and Design Guide.* IBM, 1987, 1988. Consists of [15, 16, 18, 19].

[17] MakeIndex even waits to be converted to Web, which we feel is important in spite of the growing availability of C compilers in order to allow the standard distribution tools like change files etc. to be used, cf. question 2 in [3].

[18] On JANET: `d.rhead@uk.ac.nottingham.ccc.vme`, and from other nets: `d.rhead@vme.ccc.nottingham.ac.uk`.

[18] IBM National Language Technical Center. *National Language Information: Arabic Script Languages.* Volume 3 of NLIDG [17], 1988. IBM Order Number: SE09-8003-00.

[19] IBM National Language Technical Center. *National Language Information: Hebrew.* Volume 4 of NLIDG [17], 1988. IBM Order Number: SE09-8004-00.

[20] International Business Machines Corporation. *Entering an SLSS Subscription*, 6 edition, September 1986. G320-1561-05; SLSS: System Library Subscription Service.

[21] International Business Machines Corporation. *Document Composition Facility: Generalized Markup Language. Starter Set Implementation Guide*, 5 edition, March 1988. SH35-0050-04, Release 3.2.

[22] International Business Machines Corporation. *Document Composition Facility: SCRIPT/VS Text Programmer's Guide*, 6 edition, March 1988. SH35-0069-05, Release 3.2.

[23] Knuth, Donald E[rvin]. *The TeXbook*, volume A. Addison-Wesley Publishing Company, Reading, Mass. etc., 1986. Hardcover: ISBN 0-201-13447-0, Softcover: ISBN 0-201-13448-9.

[24] Kruljac, Gabriele. "BibTeX und MAKEINDEX." *Die TeXnische Komödie. Mitgliedszeitschrift von DANTE, Deutschsprachige Anwendervereinigung TeX e.V.*, 2(1):23 – 24, 1990. Abdruck der BibTeX-Anforderungen von der europäischen TeX-Tagung in Karlsruhe, September 1989.

[25] Lamport, Leslie. *LaTeX: A Document Peparation System.* Addison-Wesley, Bonn etc., 1 edition, 1986.

[26] Mittelbach, Frank. "The doc-option." *TUGboat*, 10(2):245 – 273, July 1989.

[27] Mittelbach, Frank and Rainer Schöpf. "Towards LaTeX 3.0." In Gunther, Mary, editor, *TeX 90 Proceedings, TUGboat*, 12(1):74 – 79, Providence, March 1991. TeX Users Group. to appear.

[28] Nadrchal, J., editor. *Man-Machine Interface in the Scientific Environment. Proceedings of the 8th European Summer School on Computing Techniques in Physics. Skalský Dvůr, Czecholsovakia, 19 – 28 September 1989*, volume 61 of *Computer Physics Communications.* North Holland Publishing Company; Elsevier Science Publishers B.V., 1990.

[29] Partl, Hubert. "How to make TeX and LaTeX international." Pages 190 – 200 in *Man-Machine Interface in the Scientific Environment. Proceedings of the 8th European Summer School on Computing Techniques in Physics. Skalský Dvůr, Czecholsovakia, 19 – 28 September 1989*, J. Nadrchal [28]. Invited Paper.

[30] Partl, Hubert et al. *LaTeX-Kurzbeschreibung.* Technische Universität Wien, Wien, 1 edition, 1986.

[31] Patashnik, Oren. *Designing* BibTeX *Styles.* Stanford University, January 31 1988. BibTeX installation file: 'The part of BibTeX's documentation that is not meant for general users'.

[32] Patashnik, Oren. BibTeX*ing.* Stanford University, January 31 1988. BibTeX installation file.

[33] Rahtz, Sebastian P. Q. "Bibliographical tools." *Literary and Linguistic Computing*, 2(4):231 – 241, 1987.

[34] Rhead, David. "Towards BibTeX style-files that implement principle standards." *TeXline*, (10):2 – 8, May 1990.

[35] "TeXhax." ongoing. Modisett, Tiina and MacKay, Pierre (eds.): Moderated Network Discussion List.

[36] "TeXline. [Nr.1-7:] A newsletter of TeX users in UK and Ireland / [Nr.8ff:] A newsletter of the TeX community.." 1ff(1ff), Started 1985. Edited by Malcolm W. Clark.

[37] "TUGboat. The communications of the TeX users group." 1 ff(1 ff), 1980 ff. Barbara Beeton (ed.); subtitle for vol. 1 – 8: 'The TeX Users Group Newsletter'.

[38] University of Chicago. *The Chicago Manual of Style*, 13 edition, 1982.

[39] Wonneberger, Dr. R. *Kompaktführer LaTeX.* Addison-Wesley Kompaktführer. Addison-Wesley, Bonn etc., 2 edition, Nov. 1988. ISBN 3-89319-152-6 (VVA-Nr. 563-00152-0)

[40] Wonneberger, Reinhard. *Syntax und Exegese. Eine generative Theorie der griechischen Syntax und ihr Beitrag zur Auslegung des Neuen Testamentes, dargestellt an 2.Korinther 5,2f und Römer 3,21-26.*, volume 13 of *Beiträge zur biblischen Exegese und Theologie (BET).* Peter Lang, Frankfurt / Bern / Las Vegas, 1979.

[41] Wonneberger, Reinhard. *Leitfaden zur Biblia Hebraica Stuttgartensia.* Vandenhoeck & Ruprecht, Göttingen, 2 edition, 1986. ISBN 3-525-52180-4.

[42] Wonneberger, Reinhard. ""Verheißung und Versprechen" — A third generation approach to theological typesetting." In Désarménien, Jacques, editor, *TeX for Scientific Documentation. Second European Conference, Strasbourg, France, June [19-21], 1986. Proceedings*, number 236 in Lecture Notes in Computer Science, pages 180 – 189, Berlin / Heidelberg / London /

etc., 1986. Springer. Describes the typesetting of [49].

[43] Wonneberger, Reinhard. "TEX in an industrial environment." Preprint of [47], EDS, 1989.

[44] Wonneberger, Reinhard. "TEX yesterday, today, and tomorrow." *TEXhax*, 90(5), 7 January 1990.

[45] Wonneberger, Reinhard. "Structured document processing: the LATEX approach." Pages 177 - 189 in *Man-Machine Interface in the Scientific Environment. Proceedings of the 8th European Summer School on Computing Techniques in Physics. Skalský Dvůr, Czecholsovakia, 19 - 28 September 1989*, J. Nadrchal [28]. Invited Paper.

[46] Wonneberger, Reinhard. *Understanding BHS. A Manual for the Users of Biblia Hebraica Stuttgartensia*, volume 8 of *Subsidia Biblica*. Editrice Pontificio Istituto Biblico, Roma, 2, revised edition, 1990. ISBN 88-7653-578-0.

[47] Wonneberger, Reinhard. "TEX in an industrial environment." In *Proceedings of the 4th European TEX Conference, September 11th - 13th, 1989*, Anne Brüggemann-Klein [6]. forthcoming.

[48] Wonneberger, Reinhard. *Redaktion. Studien zur Textfortschreibung im Alten Testament, entwickelt am Beispiel der Samuelüberlieferung.* to appear.

[49] Wonneberger, Reinhard and Hans Peter Hecht. *Verheißung und Versprechen. Eine theologische und sprachanalytische Klärung.* Vandenhoeck & Ruprecht, Göttingen, 1 edition, 1986. ISBN 3-525-60367-3. Typesetting described in [42].

Index of proposals