The listbib Package*

Volker Kuhlmann†

2000/08/28

Abstract

This package typesets a listing of a (possibly large) BibTeX input file. With old implementations of texmf the string space could easily be exceeded when trying to typeset large bibliographic databases. Contemporary implementations are usually big and have much higher limits. However, listbib works with arbitrarily large BibTeX database files.

In addition to that, listbib tries to show the full content of a BibTeX file without alterations introduced by the bibliographic style used. The entry fields ISBN, ISSN, annote, library, totalpages and URL are added to the standard ones.

Contents

*This file has version number v2.2, last modified 2000/08/28.

†Email: v.kuhlmann@elec.canterbury.ac.nz. For a postal address refer to the license section.

1 License

Volker Kuhlmann, c/o University of Canterbury, ELEC Dept, Creyke Road, Christchurch, New Zealand
E-Mail: `v.kuhlmann@elec.canterbury.ac.nz`

2 History

This package is derived from `biblist`, version 1.4, 1992/01/13 by Joachim Schrod. J. Schrod's association with the university of Darmstadt seems to no longer exist. biblist is available from CTAN and is under the GNU general public license, version 1 or later.

listbib is the successor and will only work with LaTeX 2_ε. It uses the docstrip format for literate programming. The bibliography style was regenerated from btxbst.doc (unsrt), and then enhanced to be suitable for listbib. The fully commented `.bst` file is in the distribution (extension `.doc`).

This documentation has little left of the original `biblist` documentation.

3 User manual

3.1 Introduction

This package facilitates printing of large BibTeX files. With such large files — especially if the cite keys are long — the needed string space is often exceeded. Often a BigTeX is available which pushes the limits further out, but with this package any TeX will do it. Printing complete BibTeX files as they are is useful for record-keeping.

`thebibliography` The `\nocite{*}` command and `thebibliography` environment from LaTeX are used to do the work. The `thebibliography` environment is modified and eliminates huge `.aux` files and a second LaTeX-run, although cross-references used in the bibliography entries themselves may still require a second LaTeX-run.

`\bibliography` A minimal LaTeX-document must be prepared which uses the article class, loads the listbib package, and specifies the bibliography to be printed with the `\bibliography` command. Other packages can be loaded as well, for example to set margins or language-specific definitions.

`\raggedbottom` Other classes can perhaps be used, but listbib works on the assumption that `\raggedbottom` is in effect, which is why it probably won't work too well with `twocolumn` or `multicol`. A page break within an entry is not allowed and the flexibility at the page bottom is needed.

`\nocite` `\bibliographystyle` Unlike with normal LaTeX documents, use of `\bibliographystyle` is not required, although it is not prohibited. The default is the listbib.bst bibliography style. It is also not necessary to use `\nocite{*}` as that is the default. However `\nocite` can be used to select certain entries only to be printed.

A "bug" you may encounter is that `\cite` tags within BibTeX entries will not be processed. Instead the cite key itself will be printed. Note that this is not a bug, this is a feature! You have to use `\nocite` for *all* entries that shall be included in the listing. If you do not give any `\nocite` tag at all, a listing with all entries is created.

`\BibTeX` As a bonus, listbib defines the control sequence `\BibTeX` if it is not already defined. The definition is copied from `btxdoc.tex`.

3.2 Additional BibTeX Entry Fields

In addition to the entry fields known by the standard BibTeX styles, listbib recognises these fields: `ISBN`, `ISSN`, `annote`, `library`, `URL`, and `totalpages`.

The use of `ISBN` and `ISSN` should be obvious.

The `annote` field can be used for annotations or comments regarding this particular entry. The standard BibTeX styles do not make use of this feature, however other styles might. With listbib it is useful to store a short summary of the contents of this entry. As for `note`, the first word should be capitalised. Punctuation at the end of the field is added by the style if appropriate.

The `library` field is not meant to be printed in final documents, and is supplied to store e.g. the library location of a book. Useful for database listings only.

The `URL` field can be used to store a URL belonging to this bibliography entry. It is also becoming more popular to have bibliography entries which entirely consist of a URL. URLs are typeset as `\url{`⟨*URL*⟩`}`, it is recommended to use package url with documents referencing bibliography entries which contain a URL. As an additional safeguard, the listbib BibTeX-style defines `\url` to exand to ⟨*URL*⟩ if it is not already defined. This field is supported by custom-bib.

The `totalpages` field is meant to store the total number of pages of the work. This field is supported by custom-bib.

All 13 entry types take the `library`, `annote`, `totalpages` and `URL` fields. They take the fields `ISBN` and `ISSN` as well, with these exceptions: `article` only takes `ISSN`, and `unpublished` takes neither `ISBN` nor `ISSN`.

3.3 Formatting of the entries in the resulting listing

`cite key` . . . . . (Library info)

Author(s)/Editor(s).
Title.
Publication info.
Notes.

Pages: 123 • URL: `http://my.url/file`
Annotation

I.e., an open format is used. Although this needs more space I think the enhanced legibility pays back.

Note that you will not get the 'Library info' and the 'Annotation' in the above format if you use a bibliography style other than `listbib` which does not supply this information with the assumed markup. The parenthesis around the library info are produced by this style option, not by BibTeX.

3.4 Example

Suppose, there is a file `typography.bib` with references on typography.

```
% typography.bib                                    13 Jan 92
% References on Typography, Typesetting, and Book Design
%

% make sure it's at least defined:
@preamble{"\providecommand\MF{\textsc{meta-font}}"}

@book{typo:tschichold:selected,
  author = {Jan Tschichold},
  title = {Ausgew\"ahlte Aufs\"atze \"uber Fragen der Gestalt des
        Buches und der Typographie},
  publisher = {Birkh\"auser},
  address = {Basel},
  year = 1975,
  isbn = {3-7643-1946-1},
```

```
  note = {A second, unchanged, edition appeared in~1987.},
  library = {},
  annote = {A collection of his ``classic'' papers. The best book on
        typographic principles read so far. A definitive `\emph{must
        read}' for everyone interested in typography, book design,
        or typesetting.}
}

@article{typo:blostein:music,
  author = {Dorothea Blostein},
  title = {Justification of Printed Music},
  journal = cacm,
  volume = 34,
  number = 3,
  month = mar,
  year = 1991,
  pages = {88--99},
  annote = {Presents problems and solutions of the task to create a
        correct spacing while setting notes. The proposed method also
        takes the temporal relationship of notes into account.
        Handles two-dimensional layout, e.g., staffs and texts.}
}

@book{typo:sta:zapf,
  title = {Herrman Zapf and his Design Philosophy},
  publisher = {Society of Typographic Arts},
  address = {Chicago},
  year = 1987,
  isbn = {0-941447-00-6},
  note = {},
  library = {Lib MQ .B77},
  annote = {A valuable book. A chapter on his work around \MF{} is
        included. Was among the most beautiful books of the year.}
}
```

Furthermore assume that the bibliography style `listbib` is available. Then the LaTeX file

```
\documentclass{article}
\usepackage{listbib}
\begin{document}
\bibliography{typography}
\end{document}
```

may be used to create a listing with all entries. You have to run LaTeX, BibTeX, and LaTeX. (Careful readers will note that there will be a BibTeX warning about a missing author or editor in the Zapf entry.) You do *not* need to run LaTeX twice after the BibTeX run. The result looks like this:

References

`typo:tschichold:selected` . . . . . .

Jan Tschichold.
Ausgewählte Aufsätze über Fragen der Gestalt des Buches und der Typographie.
Birkhäuser, Basel, 1975. ISBN 3-7643-1946-1.
A second, unchanged, edition appeared in 1987.

> A collection of his "classic" papers. The best book on typographic principles read so far. A definitive '*must*' for everyone interested in typography, book design, or typesetting.

`typo:blostein:music` . . . . . .

Dorothea Blostein.
Justification of Printed Music.
Communications of the ACM, 34(3):88–99, March 1991.

> Presents problems and solutions of the task to create a correct spacing while setting notes. The proposed method also takes the temporal relationship of notes into account. Handles two-dimensional layout, e.g., staffs and texts.

`typo:sta:zapf` . . . . . (Lib MQ .B77)

Herrman Zapf and his Design Philosophy.
Society of Typographic Arts, Chicago, 1987. ISBN 0-941447-00-6.

> A valuable book. A chapter on his work around META-FONT is included. Was among the most beautiful books of the year.

To use a bibliography style other than `listbib`, specify it in the usual way with `\bibliographystyle`. For example, `unsrt` gives:

References

`typo:tschichold:selected` . . . . . .

Jan Tschichold.
Ausgewählte Aufsätze über Fragen der Gestalt des Buches und der Typographie.
Birkhäuser, Basel, 1975.
A second, unchanged, edition appeared in 1987.

`typo:blostein:music` . . . . . .

Dorothea Blostein.
Justification of printed music.
Communications of the ACM, 34(3):88–99, March 1991.

`typo:sta:zapf` . . . . . .

Herrman Zapf and his Design Philosophy.
Society of Typographic Arts, Chicago, 1987.

Of course, the ISBN/ISSN numbers, annotations, and library information is missing, since `unsrt` does not extract them from the database. (Careful readers

will notice another difference: The `listbib` bibliography style does not lower-case titles as the standard styles do.)

3.5 Site Configuration File

`listbib.cfg`

The listbib package will input a site configuration file `listbib.cfg` if it exists. This is a good place for changing margins or the document title, etc. The configuration file is `\input` at the end of the listbib package. Here is an example:

```
%% listbib.cfg     VK 30 Jul 2000

% Date format: 99 Month 9999
\newcommand\monthname{%
  \ifcase \month \or
    January\or February\or March\or April\or May\or June\or
    July\or August\or September\or October\or November\or December\fi
}
\def\today{%
  \number\day \space \monthname \space \number\year
}

% Latin-1 input
% (There is currently no way to do this from within a .bib file.)
\usepackage[latin1]{inputenc}
```

3.6 Inheriting Field Entries

As BibTeX currently works, missing fields in all entries which crossreference another entry are inherited from the crossreferenced entry. To illustrate, in this .bib file

```
@incollection{chapter,
    crossref="book",
    author="..", title=".."
}
@book{book,
    editor="..", booktitle="..", note=".."
}
```

the chapter entry inherits the `note` field from the `book` entry. The same is true for all other fields which can possibly be inherited. BibTeX does not make the information of whether a field was inherited available in the style file, so the inheritence can not be detected directly.

listbib implements a workaround for this which can detect inherited fields. A field is assumed to be inherited by a crossreferencing entry if its value is identical to that of the crossreferenced entry. This assumption fails when the field values are in fact identical. Please let me know of any cases where that is a problem, and I will try and fix it or make a sepaarte style which does not try to detect inherited fields. A quick fix might be to add something like a `{}` at the end of the field.

Currently inheritance is detected for the fields `note`, `annote`, `ISBN`, `ISSN`, `library`, `totalpages` and `URL`. Inherited fields are not printed in the listing.

3.7 The listbib.tex Main Document Frame

`\listbibs` The shortest way to to typeset a bibliography is to put something like this into a ⟨*file*⟩`.tex` and to run it through LaTeX, BibTeX, and (possibly more than once) LaTeX again:

```
\newcommand\listbibs{⟨bibliographies⟩}
\AtBeginDocument{⟨preamblematerial⟩}
\input{listbib}
```

For ⟨*bibliographies*⟩ list everything to be printed. Any BibTeX files containing string definitions must also be specified (first).

`\AtBeginDocument` The use of `\AtBeginDocument` is completely optional. If used, ⟨*preamblematerial*⟩ will be inserted into the preamble of `listbib.tex`.

`vmargin` `listbib.tex` also loads the packages vmargin and url if it can find them.
`url` vmargin sets up the page margins, and url provides a `\url` command for typesetting urls.

A configuration file ⟨*file*⟩`.cfg` will be loaded within the preamble of `listbib.tex`, if it exists.

This short document selects a document font size of 10pt, and double-sided printing. Unfortunately it is not possible to override that in any of the two configuration files.

3.8 The listbib Shell Script

The `listbib` program can be used to quickly generate a listing of the bibliographies given on the command line. The program is a Unix shell script; sorry if you don't use Unix. You may still be able to port it, send me a copy if you do.

Here is its usage:

```
Usage: listbib [OPTIONS] [BIBFILE[.bib] ..] BIBFILE_N[.bib]
Version VK 1.2.3, 28 Aug 2000

Options:
  -h|--help              shows help
  -o|--output FILE       generates FILE.dvi / FILE.ps (default listedbibs)
  -O|--same              as -o, but generate FILE.dvi and BIBFILE_N.ps
  -p|--ps|--postscript   generate PostScript as well as dvi
  -d|--deltemp           delete all generated files but .dvi
  -P|--psonly            generate only PostScript (delete all generated files
                           but .ps)
  --                     stop option processing; only .bib files remain
```

A temporary .tex file is created which makes use of `listbib.tex`. latex, bibtex, and possibly dvips are run on the .tex file to create the listing of all .bib files

specified. The .bib extension can be left out. Remember to specify any .bib first which define any strings needed later.

By default, the name of the .tex file is `listedbibs.tex`, this can be changed with the `-o` option. Don't specify an extension here.

To generate PostScript as well, use the `-p` option. This currently only supports dvips. I can change that if there is demand. The `-d` option will delete all temporary files when `listbib` is finished, and only leave the dvi file. `-P` will create a PostScript file and then delete all others.

To only generate a mybib.ps from mybib.bib, use `-P -O`.

An error is displayed if the .tex file to be created already exists and is not one generated by listbib.

3.9 To Do and Bugs

Inheritance is currently only checked for the fields `note`, `annote`, `ISBN`, `ISSN`, `library`, `totalpages` and `URL`. Printing of these fields is suppressed for all entry types if inherited. Should printing of other inherited fields be suppressed as well? I am less inclined to suppress `volume`, and I am not sure about `year`, `month`, etc. Please let me know what you think.

As a future enhancement one could place a word from the starting and the finishing entry on each page into the page header, like it is in a dictionary.

Some kind of section title above each .bib file would be useful.

4 Implementation

4.1 Main Document for Quick Use

The intended usage is this:

```
\newcommand\listbibs{⟨bibliographies⟩}
\input{listbib}
```

```
⟨*maindoc⟩
\documentclass[10pt,twoside]{article}
```

Set document margins if package vmargin is installed. Load package url if available; this package provides an excellent `\url` command which can be used in bibliography entries.

```
%% Packages
\IfFileExists{vmargin.sty}{% use if available
    \usepackage{vmargin}%
    \setmarginsrb{25mm}{12mm}{22mm}{8mm}{10pt}{8mm}{}{10mm}}{}
\IfFileExists{url.sty}{\usepackage{url}}{} % load if installed
%% Layout
\fboxsep 0.8pt % normally 3pt
```

And yes, we want listbib...

```
%% Package listbib
\usepackage{listbib}
```

Further document setting can be placed into a per-job configuration file, which will be loaded if found.

```
%% Load per-job configuration file if it exists
\InputIfFileExists{\jobname.cfg}{\typeout
    {*** listbib: per-job configuration file \jobname.cfg found. ***}}{}
%\csname listbibpreamble\endcsname  % call it if it is defined
```

`\listbibs` Main document part. The bibliographies to typeset are taken from `\listbibs`.

```
%% Main
\begin{document}
\bibliography{\listbibs}
\end{document}
```

⟨/maindoc⟩

4.2 Preliminary

⟨*package⟩

The main work is to supply the environment `thebibliography`; in fact, we implement it as a `description` environment. The environment has an unused argument. In addition we have to supply a correct definition for the `\bibitem` command which takes the cite key as its argument. Of course, this will result in an `\item`. After the `\bibitem` a `\library` tag may follow, with one parameter followed by a period. Then comes the reference, the (optional) annotation is a block of its own, enclosed in `\annote` and `\endannote`. After `\endannote` comes a period which should be discarded. (This should be done in the BibTeX style, but nevertheless...)

Before we start we declare some shorthands for category codes. By declaring the underscore '(_)' as letter we can use it in our macros. (I agree with D. Knuth that `\identifier_several_words_long` is more readable than `\IdentifierSeveralWordsLong` and in every case better than `\p@@@s`.) As this is a LaTeX style option the at sign is a letter anyhow; so we can use the "private" Plain and LaTeX macros; and with the underscore we can make our own macros more readable. But as we have to restore this category code at the end of this macro file we store its former value in the control sequence `\uscode`. This method is better than using a group because not all macros have to be defined global this way.

```
\chardef\uscode=\catcode`\_
\catcode`\_=11
```

4.3 The Bibliography

`thebibliography` Within an entry we use a ragged right margin. To break within an entry is always difficult and is made easier in this way. After all, the open format we use gives a

ragged impression anyhow. Within an entry we dissallow page breaks and we do not treat periods as full stops.

```
\def\thebibliography#1{%
    \description
        \rightskip \z@ plus 2em
        \interlinepenalty\@M
        \sfcode`\.=\@m
    }
\let\endthebibliography=\enddescription
```

`\bibitem` should look for the next token: If a library info exists it must go in the same line, otherwise a line break must be issued.

`\@lbibitem` `\@bibitem` But `\bibitem` itself is not of interest, it looks only for the optional argument. We have to redefine `\@bibitem`. If a bibliography style like `alpha` is used, `\bibitem` will be supplied with an optional argument. We ignore this optional argument.

After the argument(s) of `\bibitem` a line end occurs in any case. The optional `\library` tag will follow on the next line. This means we first have to gobble the line end char `^^M`, and have to check the next token afterwards. If this next token is `\library` we will do nothing since `\library` will itself end the line. Otherwise we do a line break. Experiments have shown that the dotted line at the start of every entry helps with finding the different entries.

```
\def\@lbibitem[#1]{\@bibitem}
\def\@bibitem#1{%
    \item[\texttt{\lb_sanitize{#1}}]\leavevmode
    \null \dotfill
    \begingroup
        \obeylines
        \afterassignment\prepare_check_library
        \let\next               % gobble following ^^M
    }
\def\prepare_check_library{%
        \futurelet\next\check_library
    }
\def\check_library{%
        \ifx \next\library \else \break \fi
    \endgroup
    }
\def\lb_sanitize#1{{\escapechar=-1
    \expandafter\string\csname#1\endcsname}}
```

4.4 Library

`\library` We assume that the library info and the cite key will fit into one line. If this is not true, a two line solution would have to be set up.

We add `\@highpenalty` instead of `\@M` after the library part to avoid an underfull hbox warning and an empty line if both author and editor are empty (in

which case there is a `\newblock` immediately following the `\library{..}`. Alternativelly, we could implement more `\futurelet` trickery.

```
\def\library#1{%
    \quad\textsf{\small (#1)}\penalty -\@highpenalty
    }
```

4.5 Annotation

`\newblock` A reference is divided into blocks starting with `\newblock`. Each block shall start a new line. We use `\newline` for this. A special block is the one with an annotation in it. Since we want to indent this block once more we must change the parshape. But then we have to finish the paragraph first. The annotation block starts with `\annote` and may be easily recognized in this way.

```
\def\newblock{%
    \futurelet\next\@checkAnnote
    }
\def\@checkAnnote{%
    \ifx \next\annote \else \newline \fi
    }
```

`\annote` The annotation is a paragraph on its own and indented by `\leftmargin`. Since it is not a paragraph in the logical sense we do not insert vertical glue (i.e., `\parskip`) at the top. The annotation is typeset in a smaller size. We assume that it consists of full sentences which might be made up in a proper way. Therefore we don't use a ragged margin here.

The change of `\parshape` has to be reset after the group — these changes are not local.

```
\def\annote{%
    \endgraf
    \vskip -\parsep
    \begingroup
        \reset@font\footnotesize
        \rightskip\z@skip
        \advance\linewidth -\leftmargin
        \advance\@totalleftmargin \leftmargin
        \parshape \@ne \@totalleftmargin \linewidth
    }
```

`\endannote` The period in the definition is used to gobble the period placed by BibTeX.

```
\def\endannote.{%
        \endgraf
    \endgroup
    \parshape \@ne \@totalleftmargin \linewidth
    }
```

4.6 Citation Key

\@citex

In our documents \cite will print the cite key in typewriter with a framed box around. In fact, \cite expands to \@citex, which has an optional argument. The flag \if@tempswa shows if this optional argument is there. \texttt{..} will alter \if@tempswa. So might \fbox and \ttfamily in the future. \fbox does not work inside \if, and \lb_sanitize does not work inside a \def, so we save the flag in \@cxt.

```
\def\@citex[#1]#2{\if@tempswa\def\@cxt{y}\else\let\@cxt\relax\fi
    \fbox{{\ttfamily\lb_sanitize{#2}}%
          \if y\@cxt , \ttfamily\lb_sanitize{#1}\fi}}
```

4.7 Defaults for \nocite and \bibliographystyle

\@@nocite
\@@bibliographystyle
\@@bibliography

First the original definitions of the to-be-redefined control sequences are saved. We define two macros which will issue the default given above. These macros are not private, but protected; a knowing user may redefine the 'default bibliography style' in this way.

```
\let\@@nocite=\nocite
\let\@@bibliographystyle=\bibliographystyle
\let\@@bibliography=\bibliography
```

\nocite
\bibliographystyle

If \nocite or \bibliographystyle is issued, the default tags are discarded, the original meaning is restored, and the control sequence is reissued again.

```
\def\AddNocite{\nocite{*}}
\def\AddBibliographystyle{\bibliographystyle{listbib}}
\def\nocite{%
    \let\AddNocite\relax
    \let\nocite\@@nocite
    \nocite
    }
\def\bibliographystyle{%
    \let\AddBibliographystyle\relax
    \let\bibliographystyle\@@bibliographystyle
    \bibliographystyle
    }
```

4.8 Bibliographies and Title

\bibliography

The \bibliography command will first issue the missing default tags, will produce a heading afterwards, will setup the headline, and will create the bibliography itself.

```
\def\bibliography#1{%
    \AddNocite
    \AddBibliographystyle
    \section*{\refname}%
    \mark_for_headline{#1}%
```

```
    \@@bibliography{#1}%
    }
```

`\refname` The section heading text comes from `\refname`, which we give a new default.

```
\def\refname{\BibTeX{} Database Listing}
```

`\BibTeX` Provide `\BibTeX` if it has not yet been defined. This definition is taken from btxdoc.tex (compared with LaTeX's definition of `\TeX`, this one has the "E" not touch the "T" and "X". Unfortunately the standard computer modern fonts do not supply a small caps bold, so we expect a NFSS warning when `\BibTeX` appears in the bold-faced section heading. That is, unless we do some trickery...

```
\providecommand\BibTeX{%
    \textrm{B\kern-.05em%
        {\@tempdima\f@size pt\fontsize{0.83333\@tempdima}\z@\selectfont
         I\kern-.025em B}%
        \kern-.08em T\kern-.1667em\lower.7ex\hbox{E}\kern-.125emX}%
    }
\providecommand\BibTeX{% one of the 2...
    \textrm{B\kern-.05em\textsc{i\kern-.025em b}%
        \kern-.08em T\kern-.1667em\lower.7ex\hbox{E}\kern-.125emX}%
    }
```

4.9 Page header

The page header shows the database names given with the `\bibliography` command and the current date. We do not issue a warning if no file name is given; this will be done by `\@@bibliography`. When a file name exists a comma and a space will be prepended at the very beginning of `\bib_list`. The comma is removed after the loop. When there was no file name `\bib_list` will expand to an empty list, the `\relax` serves as a catch-argument for `\@gobble`.

A `.bib` is no longer added to each bibliography filename to make the line shorter. If there are too many names the header won't fit on the page. This should be fixed somehow.

```
\def\mark_for_headline#1{%
    \def\bib_list{}%
    \@for \bib_file :=#1\do {\edef\bib_list{\bib_list, \bib_file}}%
    \edef\bib_list{\expandafter\@gobble\bib_list \relax}%
    \markboth{{\reset@font\ttfamily\bib_list}}{{\reset@font\ttfamily\bib_list}}%
    }
\def\@oddhead{\reset@font\rmfamily \hfil \leftmark \quad (\today)}
\def\@evenhead{\reset@font\rmfamily (\today)\quad \ignorespaces
    \rightmark \hfil}
```

4.10 Site Configuration File

`listbib.cfg` Further document settings can be placed into a site configuration file, which will

be loaded if found. This is a good place to put some default margin settings, for example.

```
\InputIfFileExists{listbib.cfg}{\typeout
    {*** listbib: site configuration file listbib.cfg found. ***}}{}
```

4.11 Finish

We must restore our catcode and are finished.

```
\catcode`\_=\uscode
\endinput
```

⟨/package⟩

Change History

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.