The **biblist** Style Option

Joachim Schrod[*]

January 13, 1992

1. What's this style option for?.

This style option is appropriate to create a typeset listing of a (possibly large) BibTEX input file. With such large files – especially, if the cite keys are long – the needed string space is often exceeded. Often a BigTEX is available to circumstance this problem, but with this style option each TEX will do it.

You have to prepare a LATEX document which uses the `article` style and the `biblist` style option. You may add almost all other style options, as you wish, e.g., `twoside`, `german` (or other language style options), `a4`, etc. This style option must be used with a ragged bottom; this has the effect, that it cannot be used with `twocolumn` or `multicol`.

You must issue a `\bibliography` tag which names all BibTEX databases which you want to print. You may issue a `\bibliographystyle` tag to specify how BibTEX will process its databases. (In fact, you usually must issue it since the default bibliography style is not available on most installations; see below.) You may issue `\nocite` commands if you want to print only selected entries from the databases.

2. If you have not issued the `\bibliographystyle` tag, the `biblist` bibliography style will be used. This is a non-abbreviating style which does not do any sorting and no case conversion to titles. The field `isbn` is recognized for books (and parts of), and the field `issn` is recognized for articles and such. In addition, it provides the additional fields `library` and `annote` for all entry types. The usage of these fields depends on the definition of appropriate TEX macros (which are supplied by this style option).

On interest I can provide a diff to `btxbst.doc`.

3. 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` tags 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.

Note that this style option assumes that you use either a version of LATEX 2.09 after December 1991, ILATEX, or the `babel` system. Particulary, it requires the definition of `\refname` (i.e., the title of the "references section") which is supplied by these systems. (Of course, you may supply this definition by yourself.)

[*] TU Darmstadt, Alexanderstr. 10, D-6100 Darmstadt, Germany.
Email: `schrod@iti.informatik.th-darmstadt.de`.

4. The entries in the resulting listing are formatted as follows:

`cite key` .... (Library info)
Author(s).
Title.
Publication info.
Notes.
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 your bibliography style does not supply this information with the assumed markup. (See above for the `biblist` bibliography style.) The parenthesis around the library info are produced by this style option, not by BibTeX.

5. *Example*: Suppose, there is a file `typography.bib` with references on typography.

```
% typography.bib                                13 Jan 92
%------------------------------------------------------------

%
% References on Typographie, Typesetting, and Book Design
%

% !!!! This is a kludge for biblist.doc !!!!
@preamble{"
\font\tenlogo=logo10
\let\logo=\tenlogo
\def\MF{{\logo META}\-{\logo 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 `{\it
        must\/}' 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 = {},
 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 `biblist` is available. Then the LaTeX file

```
\documentstyle[biblist]{article}

\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` ....

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 METAFONT is included. Was among the most beautiful books of the year.

Otherwise – without the `biblist` bibliography style – you must issue an appopriate tag. The following result was produced with `unsrt`:

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 numbers and the annotations are missing, since `unsrt` does not extract them from the database. (Careful readers will notice another difference: The `biblist` bibliography style does not lower titles of articles as the standard styles do.)

6. The preferred style for this style option is `article`. In fact, all styles which use `\raggedbottom` will work. This is because a page break within an entry is not allowed and the flexibility at the page bottom is needed.

7. As a future enhancement one could place an info on the starting and the finishing entry on this page in the headline, like in a dictionary.

8. This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 1, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but **without any warranty**; without even the implied warranty of **merchantability** or **fitness for a particular purpose**. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

9. This style option is supported. Send bug reports, comments and repairs.

10. Implementation.

The main work is to supply the environment thebibliography; in fact, we implement it as a description environment. The environment gets an (unused) argument. In addition we have to supply a correct expansion for the \bibitem tag 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 an own block, enclosed in \annote and \endannote. After \endannote comes a period which should be discarded. (This should be done in the BibTeX style, but nevertheless . . .)

Later we will look for the things which would not be needed, but make life easier.

11. 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 to use a group because not all macros have to be defined global this way.

```
\chardef\escape=0
\chardef\open=1
\chardef\close=2
\chardef\letter=11
\chardef\other=12
%\chardef\active=13             % is defined in Plain already

\chardef\uscode=\catcode`\_      % top level macro file!

\catcode`\_=\letter
```

12. Another care activity is needed: Users of the New Font Selection Scheme (NFSS) will perhaps change the fonts in an unexpected manner. LaTeX since December 1991 (and, of course, NFSS itself) provides the macro \reset@font to reset the font selection stuff to a defined state. Since this style might be used with older LaTeX versions (e.g., with ILaTeX) we will provide this macro.

```
\ifx \reset@font\undefined
    \let\reset@font\relax
\fi
```

13. 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
```

14. `\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.

But `\bibitem` itself is not of interest, it looks only for the optional argument. We have to redefine `\@bibitem`. If an bibliography style like `alpha` is used, `\bibitem` will be supplied an optional argument. Then it expands to `\@lbibitem`; we will define this to `\@bibitem` as well – we ignore the print label supplied by BibTeX.

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 on every entry helps finding the different entries.

```
\def\@lbibitem[#1]{\@bibitem}

\def\@bibitem#1{%
    \item[\tt #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
    }
```

15. We assume that the library info and the cite key will fit into one line. If this is not true, a two line solution must be set up.

```
\def\library#1#2.{%         #2 is only needed to gobble the period
    \quad {\sf (#1)}\break
    }
```

16. 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
    }
```

17. The annotation is an own paragraph 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` have 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
    }
\def\endannote#1.{%         #1 is only needed to gobble the period
        \endgraf
    \endgroup
    \parshape \@ne \@totalleftmargin \linewidth
    }
```

18. In our files `\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 `@tempswa` shows if this optional argument is there.

```
\def\@citex[#1]#2{\fbox{{\tt #2}\if@tempswa , #1\fi}}
```

19. Bells and Whistles.

Since the task we want to support most is the print of complete BIBTEX databases with the `biblist` bibliography style, we supply appropriate `\nocite` and `\bibliographystyle` tags, if they are missing. The `\bibliography` command shall create a heading as well.

Furthermore we set up a headline.

20. First the original meanings of the to-be-redefined control sequences are saved. We define two macros which will issue the default tags 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

\def\AddNocite{\nocite{*}}
\def\AddBibliographystyle{\bibliographystyle{biblist}}
```

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

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

22. The `\bibliography` tag will first issue the missing default tags, will produce a heading afterwards, will setup the headline, and will create the bibliography itself. For the heading we use `\refname` which should be available – either because the new LATEX, ILATEX, or the babel system is used.

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

23. The headline will consist of the database names given in the `\bibliography` tag and the current date. We ignore the database name `abbrev`, since this name is used for `@string` definitions at our site.

We do not issue a warning of 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 discarded after the loop. But when there was no file name `\bib_list` will expand to an empty list, we supply a `\relax` as a catch-up argument for `\@gobble`.

```
\def\AbbrevFileName{abbrev}

\def\mark_for_headline#1{%
    \def\bib_list{}%
    \@for \bib_file :=#1\do {%
        \ifx \bib_file\AbbrevFileName
        \else  \edef\bib_list{\bib_list, \bib_file.bib}%
        \fi
        }%
    \edef\bib_list{\expandafter\@gobble\bib_list \relax}%
    \markboth{{\reset@font\tt \bib_list}}{{\reset@font\tt \bib_list}}%
    }

\def\@oddhead{\reset@font\rm \hfil \leftmark \quad (\today)}
\def\@evenhead{\reset@font\rm (\today)\quad \ignorespaces \rightmark \hfil}
```

24. We must restore our catcode and are finished.

```
\catcode`\_=\uscode

\endinput
```