

notes2bib — Integrating notes into the bibliography*

Joseph Wright[†]

Released 2011/02/23

Abstract

The `notes2bib` package defines a new type of note, `\bibnote`, which will always be added to the bibliography. The package allows footnotes and endnotes to be moved into the bibliography in the same way. The package can be used with `natbib` and `biblatex` as well as plain LaTeX citations. Both sorted and unsorted bibliography styles are supported.

Contents

1	Introduction	2
2	Installation	2
3	Using the package	3
3.1	Auto-generated note names	4
3.2	Underlying citation system	5
3.3	Controlling the temporary database	5
3.4	Ordering notes in relation to citations	5
3.5	Converting footnotes and endnotes	6
3.6	Using unsorted bibliography styles	6
4	Data written to the aux file	7

*This file describes version v2.of, last revised 2011/02/23.

[†]E-mail: joseph.wright@morningstar2.co.uk

5	Notes for upgrading from version one	7
	Change History	8
	Index	8

1 Introduction

In most subject areas, bibliographic citations and notes are separate entities. However, in some parts of the physical sciences (chemistry and physics) it is usual for references to the literature and notes to be given together in a “References and Notes” section. By default, this requires that \LaTeX users create a notes database for each document that they write. This is also true if complex references are needed, which cannot be handled automatically.

The aim of the `notes2bib` package is to make integration of notes into the bibliography easy. Notes can be written as normal in the \LaTeX source, and are automatically moved to the bibliography. The package is compatible with sorted and unsorted bibliography styles. The package has been designed for use with numerical citations, although it will work with other systems.

2 Installation

The package is supplied in `dtx` format and as a pre-extracted zip file, `notes2bib.tds.zip`. The later is most convenient for most users: simply unzip this in your local `texmf` directory and run `texhash` to update the database of file locations. If you want to unpack the `dtx` yourself, running `tex notes2bib.dtx` will extract the package whereas `latex notes2bib.dtx` will extract it and also typeset the documentation.

The package requires \LaTeX_3 support as provided in the `expl3` and `xpackages` bundles. Both of these are available on [CTAN](#) as ready-to-install zip files. Suitable versions are available in MiKTeX 2.9 and TeX Live 2010 (updating the relevant packages online may be necessary). \LaTeX_3 , and so `notes2bib`, requires the e- \TeX extensions: these are available on all modern \TeX systems.

Typesetting the documentation requires a number of packages in addition to those needed to use the package. This is mainly because of the number of demonstration items included in the text. To compile the documentation without error, you will need the packages:

- `csquotes`

- helvet
- mathpazo
- listings

3 Using the package

The package should be loaded as normal in the preamble. The package recognises a number of options, which can also be used in the document body. These are described later in this document.

```
\usepackage[<options>]{notes2bib}
```

```
\bibnote \bibnote [<name>] {<text>}
```

The basic function provided by `notes2bib` is the `\bibnote` macro. This is used in exactly the same way as footnotes, taking a mandatory argument, the `<text>` of the note, and an optional argument, a `<name>` for the note. The `<text>` will be saved to a `LaTeX` database file for later inclusion in the bibliography, and a reference will be placed in the body text at the position of the note.

A very simple example of a bibliography note

```
\bibnote{Note for the first example}.
```

A very simple example of a bibliography note [1].

When used without the optional `<name>` argument, each note is given an automatically-generated name. If notes need to be referred to again in a document, the optional argument avoids the need to understand the detail of the automated system.

An example of a named note `\bibnote[labelled]{Note for the second example}`. The text can then continue and reference the note again later `\bibnotemark[labelled]`.

An example of a named note [2]. The text can then continue and reference the note again later [2].

Verbatim text cannot be added directly to notes (in the same way that it cannot be used in footnotes). This means that the normal care will be needed with verbatim-like material.

The next note contains some awkward text

```
\bibnote{Some \texttt{\textbackslash verb}-like output}.
```

The next note contains some awkward text [3].

`\bibnotemark` `\bibnotemark` [*⟨name⟩*]
`\bibnotetext` `\bibnotetext` [*⟨name⟩*] {*⟨text⟩*}

In common with `\footnote`, the basic `\bibnote` macro has companion macros `\bibnotemark` and `\bibnotetext`. In contrast to the LaTeX2e kernel `\footnote` macro, `\bibnote` is naturally robust and so `\bibnotemark` and `\bibnotetext` should be needed much more rarely than the `\footnote` versions.

As with the related `\footnote` functions, `\bibnotemark` can be used alone or will recognise an optional argument giving the *⟨name⟩* of the note. `\bibnotetext` also recognises the same optional *⟨name⟩* argument as well as the mandatory *⟨text⟩*.

A note without a name `\bibnotemark` can be given with some text `\bibnotetext`{Text for the fourth example}. Note can also be given names `\bibnotemark`[named], which are then used for the text `\bibnotetext`[named]{More text for the fourth example}.

A note without a name [4] can be given with some text . Note can also be given names [5], which are then used for the text.

The `\bibnotemark` macro can also be used to cross-reference notes given earlier in the document. This method is preferred for referencing notes over using the `\cite` macro as `\bibnotemark` will cope correctly with out-of-order notes (see Section 3.4).

See notes `\bibnotemark`[labelled] and `\bibnotemark`[named].

See notes [2] and [5].

`\printbibnotes` `\printbibnotes`

In most cases, there will be both notes and references in a document. The notes will be printed along with cited literature in the bibliography, produced using the `\bibliography` macro (or `\printbibliography` when using `biblatex`). However, it is possible to print only the notes in a document using the `\printbibnotes` macro.

References

[1] Note for the first example.

[2] Note for the second example.

[3] Some `\verb`-like output.

[4] Text for the fourth example.

[5] More text for the fourth example.

`\printbibnotes`

`\bibnotesetup` `\bibnotesetup` {*⟨key-value list⟩*}

The behaviour of `notes2bib` can be altered by setting one or more package options. These are given as arguments to the `\bibnotesetup` function, which takes a *⟨key-value list⟩* and uses this to set up the package. With the exception of the `file-name` option, all of the settings can be altered in the preamble or the document body. The various package options are described below.

3.1 Auto-generated note names

<code>note-name</code>	When no explicit label is given for a note, one is generated automatically by the package. This consists of two parts, a name and a number. The text of the name can be set up using the <code>note-name</code> option. This should not contain any spaces, as <code>BIBTEX</code> does not support records with spaces in names. The numerical part of the label is always generated automatically, and is the number of the note. The standard setting for <code>note-name</code> is <code>Note</code> .
<code>refsection-name</code> <code>refsection-separator</code>	When using <code>biblatex</code> 's <code>refsection</code> system, additional information is added to all labels from the second <code>refsection</code> onward (<i>i.e.</i> to all except <code>refsection 0</code>). This additional text is added before the note name, and always includes the <code>refsection</code> number. The settings <code>refsection-name</code> and <code>refsection-separator</code> determine the exact text added: <code>refsection-name</code> comes before the <code>refsection</code> number and has default <code>Refsection</code> , while <code>refsection-separator</code> comes after it and has default value <code>-</code> .

3.2 Underlying citation system

<code>cite-function</code>	<code>notes2bib</code> works by making the text of notes into citations. To do this, each note has to be "cited" in the appropriate place. <code>notes2bib</code> does not carry out any low-level citation itself: instead, a general citation macro is called. The nature of the function is set up using the <code>cite-function</code> option, which should be set to a citation macro taking one mandatory argument. The initial setting is <code>cite-function = \cite</code> .
----------------------------	---

3.3 Controlling the temporary database

<code>file-name</code>	The file name used for the database of notes is set using the <code>file-name</code> option. The standard setting is to call the file <code>notes2bib-\jobname</code> , which may not be appropriate (for example, file names containing spaces may be problematic). Setting the <code>file-name</code> option will alter the name of the file, with the file extension fixed as <code>bib</code> . In contrast to the other package options, this value can only be set in the preamble.
<code>record-type</code>	Each note is written to the database as a standard <code>BIBTEX</code> record. The type of record created is set using the <code>record-type</code> option. Usually, this will be set to <code>misc</code> ; <code>biblatex</code> v1.1a upwards provides the <code>bibnote</code> record type, and this is used if available. Some <code>BIBTEX</code> styles have dedicated support for notes: if so, the appropriate value should be set for this option.
<code>note-field</code>	The database field used to store the text of the note is available for change <i>via</i> the <code>note-field</code> option. The standard setting is <code>note</code> .
<code>keyword-entry</code>	Some styles (most notably <code>biblatex</code>) recognise keywords in <code>BIBTEX</code> records, stored in a <code>keywords</code> field. To allow the selective printing of notes, <code>notes2bib</code> includes a keyword in each note record. The keyword is set using the <code>keyword-entry</code> option: the standard setting is <code>note</code> .

3.4 Ordering notes in relation to citations

`placement` The standard method used by `notes2bib` places notes into the bibliography with no particular control of the relative position of notes with respect to citations. For unsorted bibliography styles, this will result in the notes appearing mixed in with citations. However, `notes2bib` can create notes so that they appear before or after citations, with both sorted and unsorted bibliography styles. This is controlled by the `placement` option, which recognises the values `before`, `after` and `mixed`. Setting `placement = before` places notes before citations, with the `after` setting forcing notes to appear after citation. The standard setting is `placement = mixed`, which will result in mixing of notes and citations.

When notes are placed in the bibliography without any change of order, it is possible to cross-reference to them using the standard `\cite` macro. However, when notes are out of the normal order this can lead to problems. These can be avoided by using the `\bibnotemark` macro to cross-reference notes. As this method will always work correctly, it is the recommended method for referencing notes under all circumstances.

`presort-before`
`presort-mixed`
`presort-after`
`sort-key-before`
`sort-key-mixed`
`sort-key-after` There are a number of different mechanisms used by `notes2bib` to achieve the desired ordering. Most standard `LaTeX` styles recognise a key field, which can be used to override sorting by author or title. This is used by `notes2bib`, with the note name prefixed by a string to force the sort order. The appropriate strings are stored using the options `sort-key-before`, `sort-key-mixed` and `sort-key-after`. These have standard settings `aaa`, `<blank>` and `zzz`, respectively. When the `biblatex` package is in use, a more powerful method is available to control sorting: the `presort` field. Data to be written to this field is set up using the `presort-before`, `presort-mixed` and `presort-after` options. Here, the standard values are `m1`, `mm` and `mn`, respectively. These standard values will probably be appropriate in almost all cases.

3.5 Converting footnotes and endnotes

`convert-endnotes`
`convert-footnotes` It is possible to convert both `\footnote` and `\endnote` entries in a file into `\bibnote` entries in a flexible manner. This behaviour is controlled using the `convert-endnotes` and `convert-footnotes` options; both recognise the values `true` and `false`. The original definitions for the appropriate macros are stored by `notes2bib`, and so it is possible to switch this behaviour on and off.

`\thanks` The package is designed so that converting footnotes to bibliographic notes will not affect the `\thanks` macro. Thus the option `convert-footnotes = true` can be given before `\maketitle` with no implication for and `\thanks`.

3.6 Using unsorted bibliography styles

`use-sort-key` Some bibliography styles (most notably those using the author–date system) may not work well with the settings of the package as supplied. Some of the data written by

`notes2bib` can be misunderstood by styles such as `unsrtnat`. To suppress creating a key field in the database, the option `use-sort-key` should be set to `false` with these problematic styles. At the same time, it may be necessary to alter the `note-name` option to a blank value.

```
\bibnotesetup{
  note-name      = ,
  use-sort-key = false
}
```

4 Data written to the aux file

`notes2bib` writes some information to the aux file, so that it is available between runs. The functions added to the file are not needed directly by the user, but are documented here for completeness. However, it may be necessary to worry about the aux file when splitting bibliographies.

`\recordnotes` `\recordnotes`
 When notes are placed out-of-order in in a document (using `placement = before` or `placement = after`) `notes2bib` has to record information in the current aux file. When using multiple bibliographies, this will not necessarily happen totally automatically. To force `notes2bib` to write the current out-of-order notes to the aux file, the macro `\recordnotes` is available. It should be used immediately before changing between auxiliary files.

`\TotalNotes` `\TotalNotes {<number>}`
 Records the total number of auto-numbered notes in a run. This information is needed to check if zero-filling is needed for the numbers used.

`\NotesAfterCitations` `\NotesAfterCitations {<note-list>}`

`\NotesBeforeCitations` `\NotesBeforeCitations {<note-list>}`

These functions record the notes which have been placed outside of the normal order by the package. This information is used to check for changes in note order between LaTeX runs, so that the need for re-running LaTeX and BibTeX can be detected.

5 Notes for upgrading from version one

Documents which compile with version one of `notes2bib` should work equally well with version two. The package recognises the older options and user functions (for example `\niibsetup`). These are not documented here as all new documents should use the improved structures provided here. As some auxiliary file functions have been altered, it may be necessary to delete aux files for documents processed initially with older versions of `notes2bib`.

Change History

v1.0	General: Initial public release	1	v2.ob	General: Better documentation for un-	
				packing and installation	2
v2.0	General: Second version of package us-			Improvements to details concerning	
	ing expl3 internally	1		use-sort-key option	6

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

B			
<code>\bibnote</code>	2	<code>note-name</code> 4	
<code>\bibnotemark</code>	3	<code>placement</code> 5	
<code>\bibnotesetup</code>	4	<code>presort-after</code> 5	
<code>\bibnotetext</code>	3	<code>presort-before</code> 5	
		<code>presort-mixed</code> 5	
C		<code>record-type</code> 4	
<code>cite-function</code> (option)	4	<code>refsection-name</code> 4	
<code>convert-endnotes</code> (option)	5	<code>refsection-separator</code> 4	
<code>convert-footnotes</code> (option)	5	<code>sort-key-after</code> 5	
		<code>sort-key-before</code> 5	
F		<code>sort-key-mixed</code> 5	
<code>file-name</code> (option)	4	<code>use-sort-key</code> 6	
K			
<code>keyword-entry</code> (option)	5	P	
		<code>placement</code> (option)	5
N		<code>presort-after</code> (option)	5
<code>note-field</code> (option)	5	<code>presort-before</code> (option)	5
<code>note-name</code> (option)	4	<code>presort-mixed</code> (option)	5
<code>\NotesAfterCitations</code>	6	<code>\printbibnotes</code>	3
<code>\NotesBeforeCitations</code>	6		
O		R	
options:		<code>record-type</code> (option)	4
<code>cite-function</code>	4	<code>\recordnotes</code>	6
<code>convert-endnotes</code>	5	<code>refsection-name</code> (option)	4
<code>convert-footnotes</code>	5	<code>refsection-separator</code> (option)	4
<code>file-name</code>	4		
<code>keyword-entry</code>	5	S	
<code>note-field</code>	5	<code>sort-key-after</code> (option)	5
		<code>sort-key-before</code> (option)	5
		<code>sort-key-mixed</code> (option)	5

T

U

`\thanks` 6
`\TotalNotes` 6 `use-sort-key (option)` 6