

# The `collref` Package\*

Niklas Beisert

Institut für Theoretische Physik  
Eidgenössische Technische Hochschule Zürich  
Wolfgang-Pauli-Strasse 27, 8093 Zürich, Switzerland

[nbeisert@itp.phys.ethz.ch](mailto:nbeisert@itp.phys.ethz.ch)

17 January 2018, v2.0c

## Abstract

`collref` is a  $\text{\LaTeX}2_{\epsilon}$  package to automatically collect multiple `\bibitem` references which always appear in the same sequence in `\cite` into a single `\bibitem` block.

## Contents

<b>1 Introduction</b>	<b>1</b>
<b>2 Usage</b>	<b>2</b>
<b>3 Information</b>	<b>3</b>
3.1 Copyright . . . . .	3
3.2 Files and Installation . . . . .	3
3.3 Related CTAN Packages . . . . .	4
3.4 Revision History . . . . .	4
<b>A Sample File</b>	<b>5</b>
<b>B Implementation</b>	<b>6</b>

## 1 Introduction

Suppose a manuscript uses the following set of four references:

- [1] Reference A
- [2] Reference B
- [3] Reference C
- [4] Reference D

Now if references B and C cover similar or related material, they might always be cited together as in “[. . . , 2, 3, . . .]” throughout the manuscript. In some (physics) journals it is then customary to collect the two references into a single reference

---

\*Earlier versions developed at: Max-Planck-Institut für Gravitationsphysik (Albert-Einstein-Institut), Potsdam, Germany; AEI-2009-054

- [1] Reference A
- [2] Reference B  
Reference C
- [3] Reference D

and cite it by “[... , 2, ...]”. The package `collref` automates this process by analysing the `\cite` commands and identifying blocks of references which always appear in conjunction. These blocks are collapsed to a single item in the bibliography. Please note that `collref` requires the sequence of `\bibitem` entries to match with the sequence of `\cite` blocks. This is most easily achieved through the use of `BIBTEX` with any *unsorted* style.

## 2 Usage

**Inclusion.** To use `collref` simply add the command

$$\backslash usepackage\{collref\}$$

to the preamble of your `LATEX` document. No further interaction is required.

**Punctuation.** `collref` provide basic punctuation between collected references. This is specified through the package options `\usepackage[opt]{collref}` where *opt* is one of the following

nosep (default)	parsep	bulletsep	punctsep
no separator:	separated by <code>\par</code> :	separated by ‘•’:	punctuated by ‘;’ and ‘.’:
[1] A	[1] A	[1] A	[1] A .
[2] B C	[2] B C	[2] B • C	[2] B ; C .
[3] D	[3] D	[3] D	[3] D .

cf. note on spacing below.

Alternative separators can be specified in the preamble through the command:

$$\backslash collectsep[punctuation]\{separator\}$$

The *separator* appears between references in a block and the *punctuation* at the end of a block of references.

**Bibliography Preparation.** Please note that only such blocks of references can be collapsed which appear in the same order for `\cite` commands as for `thebibliography`. It is recommended to prepare the bibliography through `BIBTEX` which does this automatically. You must use a style which does not sort the references but preserves the order in which they were `\cite`d, e.g. `unsrt.bst`.

Also note that `collref` suppresses new paragraphs invoked by empty lines in the bibliography. This allows to use standard `BIBTEX` styles which commonly separate reference entries by empty lines. If these empty lines would be expanded to new paragraphs, `collref` would not be able to join two references properly. Therefore new paragraphs have to be invoked by the command `\par`.

If you wish to use the style `punctsep`, please refer to the following note on spacing and punctuation.

**Spacing and Punctuation.** References are usually punctuated in some way. Three of the predefined styles – `nosep`, `parsep` and `bulletsep` – preserve the punctuation from the bibliography.

The fourth predefined style – `punctsep` – automatically performs the punctuation. This however requires care in the preparation of the bibliography: The entries have to be provided *without* punctuation. Furthermore, there must not be *whitespaces* at the end of an entry. They can be suppressed with ‘%’ or `\ignorespaces` directly following the last word of the entry. See Appendix A for an example. The standard BIB<sub>T</sub>E<sub>X</sub> styles, e.g. `unsrt.bst`, have to be adjusted to remove the punctuation and whitespaces.

**Control.** The package `collref` provides one command to control which references (not) to collect:

```
\nocollect{label}
```

It ensures that the label *label* starts a new block of references. It is not collapsed with earlier references. Later references, however, can still be collapsed to the end of *label*.

**Labels for Blocks of References.** While `collref` aims to automatically collect similar references into a single block, it is often convenient for the author to refer to such blocks with a single citation label. Standard T<sub>E</sub>X/L<sub>A</sub>T<sub>E</sub>X commands can be used to define such a block:

```
\newcommand{\blocklabel}{label1, label2, ...}
```

Subsequently this block can be referenced with `\cite{..., \blocklabel, ...}`.

## 3 Information

### 3.1 Copyright

Copyright © 2003–2018 Niklas Beisert

This work may be distributed and/or modified under the conditions of the L<sub>A</sub>T<sub>E</sub>X Project Public License, either version 1.3 of this license or (at your option) any later version. The latest version of this license is in <http://www.latex-project.org/lppl.txt> and version 1.3 or later is part of all distributions of L<sub>A</sub>T<sub>E</sub>X version 2005/12/01 or later.

This work has the LPPL maintenance status ‘maintained’.

The Current Maintainer of this work is Niklas Beisert.

This work consists of the files `README.txt`, `collref.ins` and `collref.dtx` as well as the derived files `collref.sty`, `collsamp.tex` and `collref.pdf`.

### 3.2 Files and Installation

The package consists of the files

<code>README.txt</code>	readme file
<code>collref.ins</code>	installation file
<code>collref.dtx</code>	source file
<code>collref.sty</code>	package file
<code>collsamp.tex</code>	sample file
<code>collref.pdf</code>	manual

The distribution consists of the files `README.txt`, `collref.ins` and `collref.dtx`.

- Run (pdf)L<sup>A</sup>T<sub>E</sub>X on `collref.dtx` to compile the manual `collref.pdf` (this file).
- Run L<sup>A</sup>T<sub>E</sub>X on `collref.ins` to create the package `collref.sty` and the sample `collsamp.tex`. Copy the file `collref.sty` to an appropriate directory of your L<sup>A</sup>T<sub>E</sub>X distribution, e.g. `texmf-root/tex/latex/collref`. Alternatively, you may copy `collref.sty` to the local directories of manuscripts for which you wish to use `collref`.

### 3.3 Related CTAN Packages

The objective and some of the implementation of the `collref` package is similar to the CTAN packages `mcite` by Thorsten Ohl and `mciteplus` by Michael Shell, but the functionality is different in several respects:

- `collref` is intended to work transparently: L<sup>A</sup>T<sub>E</sub>X documents which compile with `collref` should also compile fine without invoking `collref` (obviously without collected references). The package decides automatically which references can be collapsed, no further interaction of the author is required.

`mcite` and `mciteplus` leave the decision/duty to collapse certain references using the modified syntax `\cite{A,*B,*C}`.

- `mcite` and `mciteplus` are intended to handle punctuations in collapsed references correctly. This requires a specialised B<sub>I</sub>B<sub>T</sub><sub>E</sub>X style. No effort is made in `collref` in this regard. Some minor modification in `collref.sty` together with a modified B<sub>I</sub>B<sub>T</sub><sub>E</sub>X style might achieve basic punctuation features similar to `mcite`.

The package `collref` has been tested with other CTAN packages concerned with citations and the bibliography:

- `collref` works in conjunction with `cite`. Note that you must load `cite` *before* `collref` so that the latter can pass the correctly reduced list of references down to `cite`. Tested with v5.1 and v5.3 (2010/09/10).
- `collref` works in conjunction with `hyperref`. The two packages can be loaded in any sequence. Tested with v6.78s and v6.83m (2012/11/06).

### 3.4 Revision History

**v2.0c:** 2018/01/17

- manual rearranged

**v2.0b:** 2014/08/31

- updated author addresses
- minor internal changes

**v2.0:** 2009/09/07

- proper punctuation added (thanks to Oleg Zhiron for suggestion)
- blocks of references enabled (thanks to Oleg Zhiron for suggestion)
- manual extended

**v1.0:** 2009/06/09

- streamlined detection of chains
- manual and installation package added
- renamed package to collref due to name clash on CTAN
- first version published on CTAN

**v0.9:**

- package named collect; unpublished

## A Sample File

In this section we provide a sample file.

```
1 \documentclass{article}
2
3 %%\usepackage{cite}
4 \usepackage[punctsep]{collref}
5 %%\usepackage{hyperref}
6
7 \begin{document}
8
9 \def\tworef{c8,c9}
10
11 \cite{c1,c2,c3,c4}
12 \nocollect{c3}
13 \cite{c5,c6,c7,\tworef}
14 \cite{c5,c6,c7}
15 \cite{c7,\tworef}
16
17 \begin{thebibliography}{11}
18 \bibitem{c1} reference 1%
19 \bibitem{c2} reference 2%
20 \bibitem{c3} reference 3%
21 \bibitem{c4} reference 4%
22 \bibitem{c5} reference 5
23 \bibitem{c6} reference 6
24 \bibitem{c7} reference 7 %
25 \bibitem{c8} reference 8\ignorespaces
26 \bibitem{c9} reference 9\ignorespaces
27 \end{thebibliography}
28
29 \end{document}
```

It produces the output:

[1, 2] [3, 4, 5] [3, 4] [4, 5]

- [1] reference 1; reference 2.
- [2] reference 3; reference 4.
- [3] reference 5 ; reference 6 .
- [4] reference 7 .

[5] reference 8; reference 9.

Note the different behaviour for references 5, 6 and 7 for which trailing whitespaces were not removed.

## B Implementation

In this section we describe the package `collref.sty`.

**Internal Lists.** For each bibliography label *label* the package maintains a predecessor `\nc@p@label` and a successor `\nc@s@label`. These are initially undefined. When a label *label* is first cited these labels are set to the *predecessor* and *successor* labels, respectively, in `\cite{... , predecessor, label, successor, ...}`. An empty `\nc@p@label` or `\nc@s@label` refers to the beginning and end of a block, respectively. Whenever `\cite` finds conflicting blocks (non-matching predecessors or successors in two `\cite`'s), it terminates the blocks to the maximum common overlap.

**Interface.** The package provides two public commands, described above:

```
30 \newcommand{\collectsep}[2] [] {\def\nc@punct{#1}\def\nc@sep{#2}}
31 \newcommand{\nocollect}[1] {\nc@breakbefore{#1}\ignorespaces}
```

**Package Options.** The package provides four predefined separators described above:

```
32 \DeclareOption{nosep}{\collectsep{}}
33 \DeclareOption{parsep}{\collectsep{\par}}
34 \DeclareOption{bulletsep}{\collectsep{\textbullet{ } }}
35 \DeclareOption{punctsep}{\collectsep[.]{; }}
36 \ExecuteOptions{nosep}
37 \ProcessOptions
```

**Internal Commands.** Some internal commands for abbreviation:

```
38 \newcommand{\nc@getcsname}[1] {\csname #1\endcsname}
39 \newcommand{\nc@setcsname}[2] {\expandafter\xdef\csname #1\endcsname{#2}}
```

Command to terminate the chain before a label: The predecessor of the label is terminated. If the predecessor was active, its successor is also terminated.

```
40 \newcommand{\nc@breakbefore}[1] {%
41   \edef\nc@citepred{\@ifundefined{nc@p@#1}{\nc@getcsname{nc@p@#1}}}%
42   \ifx\nc@citepred\@empty\else\nc@setcsname{nc@s@\nc@citepred}{\fi%
43   \nc@setcsname{nc@p@#1}{}%
44 }
```

Command to terminate the chain after a label. Similar to the above command.

```
45 \newcommand{\nc@breakafter}[1] {%
46   \edef\nc@citesucc{\@ifundefined{nc@s@#1}{\nc@getcsname{nc@s@#1}}}%
47   \ifx\nc@citesucc\@empty\else\nc@setcsname{nc@p@\nc@citesucc}{\fi%
48   \nc@setcsname{nc@s@#1}{}%
49 }
```

**Citations.** Hack for `\@citex`: It is assumed that (as in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>) `\cite` eventually passes down to `\@citex`.

```

50 \let\nc@old@citex\@citex
51
52 \def\@citex[#1]#2{%
53   \let\nc@citecomma\@empty%
54   \let\nc@citestring\@empty%
55   \let\nc@citelast\@empty%
56   \edef\nc@citelist{#2}%

```

Main loop to process the arguments of `\cite`. The current label is stored in `\nc@citethis`.

```

57   \@for\nc@citethis:={\nc@citelist}\do{%
58     \edef\nc@citethis{\expandafter\@firstofone\nc@citethis\@empty}%

```

The first entry has no predecessor, terminate the chain.

```

59     \ifx\nc@citelast\@empty%
60       \nc@breakbefore{\nc@citethis}%
61     \else%

```

Non-first entry: Fill undefined successor and predecessors entries with the current chain sequence.

```

62       \@ifundefined{nc@s@\nc@citelast}%
63         {\nc@setcsname{nc@s@\nc@citelast}{\nc@citethis}}{}%
64       \@ifundefined{nc@p@\nc@citethis}%
65         {\nc@setcsname{nc@p@\nc@citethis}{\nc@citelast}}{}%

```

Get the successor and predecessors for the last and current entry, respectively.

```

66       \edef\nc@citesucc{\nc@getcsname{nc@s@\nc@citelast}}%
67       \edef\nc@citepred{\nc@getcsname{nc@p@\nc@citethis}}%

```

In case of mismatching chains: terminate all links.

```

68       \ifx\nc@citesucc\nc@citethis%
69         \ifx\nc@citepred\nc@citelast%
70           \else%
71             \nc@breakafter{\nc@citelast}%
72             \nc@breakbefore{\nc@citethis}%
73           \fi%
74         \else%
75           \nc@breakafter{\nc@citelast}%
76           \nc@breakbefore{\nc@citethis}%
77         \fi%
78       \fi%

```

Get content of `\b@label` entry to find out whether the `\bibitem{label}` entry exists. We need to take special care of extended label definitions in `hyperref`.

```

79       {\def\hyper@@link[##1]##2##3##4{##4}%
80        \xdef\nc@citelabel{\nc@getcsname{b@\nc@citethis}}}%

```

Only add those labels which actually exist to the pass-on string. This removes collapsed references from the citation marks.

```

81       \ifx\nc@citelabel\@empty\else%
82         \edef\nc@citestring{\nc@citestring\nc@citecomma\nc@citethis}%
83       \fi%

```

Write `\citation` tag to .aux file in original order. Some duplicate `\citation`'s will be written by the original `\citex` code, but these will have no impact.

```
84 \if@filesw\immediate\write\@auxout{\string\citation{\nc@citethis}}\fi%
```

Continue to next label.

```
85 \edef\nc@citelast{\nc@citethis}%
86 \def\nc@citecomma{,}%
87 }%
```

The last entry has no successor, terminate the chain.

```
88 \nc@breakafter{\nc@citelast}%
```

Pass on to original L<sup>A</sup>T<sub>E</sub>X code.

```
89 \nc@old@citex[#1]{\nc@citestring}%
90 }
```

**Bibliography.** Enhance the `thebibliography` environment to a) set the `\nc@biblast` label to something, and empty `\nc@nextpunct` (no predecessor for the first entry), b) convert linebreaks into whitespaces (avoid implicit `\par`'s), and c) put the final punctuation for the last entry.

```
91 \let\nc@old@thebibliography\thebibliography
92 \let\nc@old@endthebibliography\endthebibliography
93
94 \def\thebibliography{%
95 \xdef\nc@biblast{asldjfhasklfh}%
96 \xdef\nc@nextpunct{}%
97 \catcode'\^M=10%
98 \nc@old@thebibliography}
99
100 \def\endthebibliography{%
101 \nc@nextpunct%
102 \nc@old@endthebibliography}
```

Overwrite `\bibitem`: It is assumed that the native L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> code is equivalent but with the L<sup>A</sup>T<sub>E</sub>X internals `\@lbibitem` and `\@bibitem`. Some other packages may also redefine `\bibitem` and this will inevitable cause compatibility issues. This implementation is safe with current versions of `hyperref`.

```
103 \def\bibitem{\@ifnextchar[\nc@lbibitem\nc@bibitem]}
```

`\nc@noitem` is invoked in place of the original `\@bibitem` or `\@lbibitem` for collapsed references:

```
104 \def\nc@noitem#1{%
105 \if@filesw\immediate\write\@auxout{\string\bibcite{#1}{}}\fi%
106 \ignorespaces}
```

The hack for `\@bibitem`: It checks whether this reference is part of a block. If so, put separator and collect by `\nc@noitem`. Otherwise put punctuation and pass down to `\@bibitem`. Finally let `\nc@biblast` point to current item, and fill the punctuation `\nc@nextpunct` for the next entry.

```
107 \def\nc@bibitem#1{%
108 \edef\nc@bibpred{\@ifundefined{nc@p@#1}{\nc@getcurname{nc@p@#1}}}%
109 \ifx\nc@biblast\nc@bibpred\nc@sep\nc@noitem{#1}%
110 \else\nc@nextpunct\@bibitem{#1}\fi%
```

```
111 \xdef\nc@biblast{#1}%
112 \xdef\nc@nextpunct{\nc@punct}%
113 \ignorespaces}
```

Similar hack for @lbibitem:

```
114 \def\nc@lbibitem[#1]#2{%
115 \edef\nc@bibpred{\ifundefined{nc@p@#2}{\nc@getcurname{nc@p@#2}}}%
116 \ifx\nc@biblast\nc@bibpred\nc@sep\nc@noitem{#2}%
117 \else\nc@nextpunct\@lbibitem[#1]{#2}\fi%
118 \xdef\nc@biblast{#2}%
119 \xdef\nc@nextpunct{\nc@punct}%
120 \ignorespaces}
```