mcite:
Multiple Citations on One Key

Thorsten Ohl*

Technische Hochschule Darmstadt
Schloßgartenstr. 9
D-64289 Darmstadt
Germany

April 4, 1996

Abstract
This LATEX 2ε package provides support for collapsing multiple cita-
tions into one, as customary in physics journals.

Contents
1 Introduction 1

2 Implementation 3
  2.1 Collecting Information ........................................ 3
  2.2 Producing the Output ........................................ 5

A Driver File 6

B Copying 7

1 Introduction
Imagine we want to refer to the standard model [1] of particle physics. Instead
of the standard BiBTeX [2] format


and Wiksell.


*e-mail: Thorsten.Ohl@Physik.TH-Darmstadt.de
it is customary in physics journals to collapse such references into one


The slight typographyical changes are easily taken care of by changing the \texttt{BibTeX} style, i.e. by using \texttt{phaip.bst}. The hard part is to convince \texttt{BibTeX} to collapse the entries. \texttt{mcite} takes care of that. 

\texttt{mcite} takes care of that. This package overloads the \LaTeX \cite command. All keys starting with a \texttt{*} (which should be sufficiently unlikely in a \texttt{BibTeX} key) will be added to the previous item. For example the following reference \cite{texbook,latex,*latex-companion,Glashow,*Salam,*Weinberg} was created by \cite{texbook,latex,*latex-companion,Glashow,*Salam,*Weinberg}.

\textbf{Hacking the \texttt{BibTeX} Style}

With the standard \texttt{BibTeX} styles, it is almost impossible to provide proper punctuation (i.e. “;” between collapsed items). In principle one could play tricks with an active “;”, but there are too many special cases in real life \texttt{BibTeX} entries that can go wrong. Fortunately, it is fairly easy to customize \texttt{BibTeX} styles for our purposes.

Almost all \texttt{BibTeX} styles known to (wo)mankind contain a function

\begin{verbatim}
FUNCTION {fin.entry}
{ add.period$ write$ newline$}
\end{verbatim}

removing the \texttt{add.period$} and escaping the \texttt{newline$} with \texttt{\relax} will take care of the paragraph breaks and allow \texttt{mcite} to provide the proper punctuation:

\begin{verbatim}
FUNCTION {fin.entry}
{ write$ "\relax" write$ newline$ "\relax" write$}
\end{verbatim}

Note that \texttt{%} would be almost as good as \texttt{\relax} but the latter also works if the \texttt{\catcode} of \texttt{%} has been changed.

In the function \texttt{begin.bib}, the \texttt{\begin{thebibliography}} should be replaced by \texttt{\begin{mcbibliography}}. Similary, in \texttt{end.bib} the \texttt{\end{thebibliography}} by \texttt{\end{mcbibliography}}. The \texttt{mcbibliography} environment will inform \texttt{mcite} that it has to provide the punctuation.

\textbf{Bugs and Deficiencies}

The following bugs and deficiencies are known:

1. \texttt{mcite} only works with unsorted bibliography styles. Without modifications to \texttt{BibTeX}, it seems impossible to support sorted bibliography styles.

2. \texttt{mcite}’s \texttt{\cite} is less forgiving about bogus spaces. This can be fixed.
2 Implementation

It’s good practice to identify this version of this package. We do this by parsing an RCS Id string and storing the result in the conventional TeX control sequences:

\begin{verbatim}
\langle∗package⟩
\def\fileversion{v0.04}
\NeedsTeXFormat{LaTeX2e}
\def\RCS#1#2\endRCS{% 
  \ifx$#1% 
    \@RCS $#2 \endRCS
  \else 
    \@RCS $*: #1#2$ \endRCS
  \fi}%
\def\@RCS $#1: #2,v #3 #4 #5 #6$ \endRCS{% 
  \gdef\filename{#2}% 
  \gdef\filerevision{v#3}% 
  \gdef\filedate{#4}%
\RCS $Id: mcite.dtx,v 1.6 1996/01/01 23:52:29 ohl Exp \endRCS}$
\end{verbatim}

And now the standard procedure:

\begin{verbatim}
\ProvidesPackage{mcite}[%filedate\space multiple citations]
\typeout{Package: 'mcite'\fileversion\space <\filedate> (tho) PRELIMINARY TEST RELEASE}
\wlog{English documentation \docdate\space (tho)\}
\end{verbatim}

2.1 Collecting Information

\begin{verbatim}
\def\@enamedef#1{\expandafter\edef\csname #1\endcsname}
\def\mc@single#1{\global\@enamedef{mc*sg*#1}{}%}
\def\mc@head#1#2{\global\@enamedef{mc*hd*#1}{#2}}
\def\mc@tail#1#2{\global\@enamedef{mc*tl*#1}{#2}}
\def\mc@ifsingle#1#2#3{\@ifundefined{mc*sg*#1}{#3}{#2}}
\def\mc@ifhead#1#2#3{\@ifundefined{mc*hd*#1}{#3}{#2}}
\def\mc@iftail#1#2#3{\@ifundefined{mc*tl*#1}{#3}{#2}}
\def\mc@thehead#1{\@nameuse{mc*tl*#1}}
\def\mc@thetail#1{\@nameuse{mc*hd*#1}}
\end{verbatim}

\mcite We are overloading the original \cite command such that \cite{latex,*latex-companion,texbook} will give \cite{3, 4}.

\begin{verbatim}
\let\orig@cite\cite
\def\mcite{%
  \@ifnextchar[%
    {\PackageWarning{mcite}%(optional argument to \protect\cite\space not supported)}%
  \@tempswatrue
  \expandafter\mc@citex\mc@gobbleopt}%
  \@tempswatrue
  \mc@cite}
\end{verbatim}

Gobble up all optional arguments

\begin{verbatim}
\def\mc@gobbleopt[#1]{%}
\end{verbatim}

\mcite
\cite If we postpone the overloading of \cite until the \begin{document}, we can coexist with cite.sty. If we were to do it now, we would have to demand that cite.sty is loaded before mcite.sty.
\begin{verbatim}
39 \AtBeginDocument{\let\cite\mcite}
\end{verbatim}

\mc@cite First of all: expand the argument:
\begin{verbatim}
40 \def\mc@cite#1{%
41   \edef\mc@temp{#1}%
42   \expandafter\mc@cite@\expandafter{\mc@temp}
}\end{verbatim}
Here we do the actual “preprocessing” of the \cite keys.
\begin{verbatim}
43 \def\mc@cite@#1{%
44   \mc@firsttrue
45   \@for\mc@@@:=#1\do{%
46     \ifx*#1
47       \PackageWarning{mcite}{}
48       \mc@firstfalse
49       \edef\mc@list{\mc@key}{}
50       \else
51       \mc@dottail
52       \fi
53   \else
54       \mc@dohead
55   \fi
56 }%
57 \expandafter\orig@cite\expandafter{\mc@list}
\end{verbatim}
Write a \citation command to the .aux file in any case. This gives some duplication but keeps things in order.
\begin{verbatim}
59 \def\mc@dohead{%
60   \mc@ifhead{\mc@key}%
61     {\PackageWarning{mcite}{}
62      \{head `\mc@key' already used as tail of `\mc@thehead{\mc@key}'}%}
63   {}}
64 \def\mc@dottail{\mc@key}%
65 \edef\mc@curhead{\mc@key}%
66 \if\mc@first
67 \mc@firstfalse
68 \edef\mc@list{\mc@key}%
69 \else
70 \edef\mc@list{\mc@list, \mc@key}%
71 \fi
72 \def\mc@dottail{%
73 \mc@ifhead{\mc@key}%
74     {\PackageWarning{mcite}{}
75      \{tail `\mc@key' already used as head}{}
76   {}}
77 \mc@tail{\mc@key}{\mc@curhead}
78 \def\mc@ifstar#1#2\sentinel#3#4{%
79 \ifx*#1%
80 \end{verbatim}

\cite
## 2.2 Producing the Output

The simplest approach is to just drop the `\bibitem` if we’re processing a continuation (it is just an `\item` with associated `.aux` file action):

```latex
\let\orig@bibitem\@bibitem
\def\@bibitem#1{\orig@bibitem{#1}}
```

If the BibTeX style supports `mcite`, we have to provide punctuation ourselves.

```latex
\ifmc@bstsupport
\mc@iftail{#1}{\orig@bibitem{#1}}
\mc@firstfalse
\else
\mc@iftail{#1}{\orig@bibitem{#1}}
\fi
```

The `mcbibliography` environment instructs `mcite` to provide its own punctuation.

```latex
\def\mcbibliography{%\mc@bstsupporttrue\mc@firsttrue\thebibliography}
\def\endmcbibliography{%\mc@bstsupportfalse\thebibliography}
```

That’s it!

### References


Oren Patashnik. Designing BIBTEX styles. The part of BIBTEX’s documentation that’s not meant for general users, 8 February 1988.


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.

A Driver File

In general we will of course format this documentation with itself.
\documentclass[a4paper]{article}
\usepackage{doc}
\usepackage{mcite}
\EnableCrossrefs
\RecordChanges
\CodelineIndex
\makeatletter
\def\bitem#1{\item[\@biblabel{#1}]}
\makeatother
\def\MCITE/{\texttt{mcite}}
\DoNotIndex{\advance, \baselineskip, \batchmode, \begin, \catcode, \char}
\DoNotIndex{\def, \divide, \docdate, \DocInput}
\DoNotIndex{\documentclass, \else, \emph, \EnableCrossrefs, \end, \endcsname}
\DoNotIndex{\errmessage, \errorstopmode, \expandafter, \fbox, \fi, \filedate}
\DoNotIndex{\filename, \fileversion, \font, \fontencoding, \fontfamily}
\DoNotIndex{\fontseries, \fontshape, \fontsize, \gdef, \goodbreak, \hfil}
\DoNotIndex{\hoffset, \hskip, \ifcase, \ifx, \InputIfFileExists, \item, \let}
\DoNotIndex{\line, \LoadClass, \makebox, \mbox, \multiply, \NeedsTeXFormat}
\DoNotIndex{\newdimen, \newenvironment, \newif, \nobreak, \normalfont}
\DoNotIndex{\parindent, \parskip, \PassOptionsToClass, \ProcessOptions}
\DoNotIndex{\parindent, \parskip, \PassOptionsToClass, \ProcessOptions}
\DoNotIndex{\ProvidesClass, \put, \quad, \raisebox, \RecordChanges, \relax, \rule}
\DoNotIndex{\selectfont, \space, \ss, \texttt, \textwidth, \the, \thispagestyle}
\DoNotIndex{\topmargin, \usepackage, \voffset, \vskip, \vspace, \wlog}
\begin{document}
\bibliographystyle{unsrt}%%%{physics}
\DocInput{mcite.dtx}
\end{document}

\section*{B Copying}

\texttt{mcite} 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 2, or (at your option) any later version.

\texttt{mcite} is distributed in the hope that it will be useful, but \textit{without any warranty}; without even the implied warranty of \textit{merchantability} or \textit{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.