L'extension index *

David M. Jones

28/09/1995

1 Introduction

✖This style file reimplements LaTeX's indexing macros to provide better and more robust support for indexes. In particular, it provides the following features : [1]

1. Support for multiple indexes.

2. Indexing of items by counters other than the page number.

3. A ∗-variant of the `\index` command that, in addition to putting it's argument in the index, also typesets it in the running text.

4. The `showidx` style option has been merged into this file. The command `\proofmodetrue` can be used to enable the printing of index entries in the margin of pages. The size and style of font can be controlled with the `\indexproofstyle` command.

5. A two-stage process, similar to that used to create tables of contents, for creating the raw index files. This means that when processing a portion of a document using the `\includeonly` command, the index entries from the rest of the document are not lost.

6. A more robust `\index` command. In particular, it no longer depends on `\catcode` changes to work properly, so the new `\index` command can be used in places that the original couldn't, such as inside the arguments of other macros.

2 Creating an index with LaTeX

Conceptually, there are four stages to creating an index. First, LaTeX must be informed of your intention to include an index in your document. Second, you must add appropriate markup commands to your document to tell LaTeX what to

---

* Ce fichier a pour numéro de version v4.1beta et a été mis à jour le 28/09/1995. Son titre original est « *A new implementation of LaTeX's indexing commands* ». La version définitive de ce fichier se trouve à l'adresse `ftp://theory.lcs.mit.edu/pub/tex/index/`.

1. Earlier versions of this package provided a "shortindexing" feature (see below for description). This feature is now deprecated and will be removed in a future release of this package.

put in the index. Third, after LaTeX has been run on your document, the raw index information must be processed and turned into a form that LaTeX can process to typeset the index. Finally, the finished index must be inserted at the appropriate point in your document.

In LaTeX, these steps are accomplished with the commands `\makeindex`, `\index`, `\printindex`, and (typically) with the auxiliary program MakeIndex. For example, assuming that your main file is called `foo.tex`, `\makeindex` opens the file `foo.idx` and initializes it for holding the raw index entries, and `\index` is used to add raw index entries into `foo.idx`. Then the raw index file is processed by MakeIndex, which puts the finished index in `foo.ind`. Finally, the `\printindex` command is used in your LaTeX document to indicate where the file `foo.idx` should be inserted, i.e., where the index should appear in your document.

The `index` package modifies the `\makeindex`, `\index`, and `\printindex` commands, as described below.

3   The user interface

There are four pieces of information associated with each index :

1. A short, unique tag that identifies the index.

2. The extension of the output file where the raw index information will be put by LaTeX.

3. The extension of the input file where the processed information created by MakeIndex will be stored to be read in later by LaTeX.

4. The title of the index.

`\newindex`   Correspondingly, the `\newindex` command has four required arguments. For example, to declare an author index, you might use the following :

```
\newindex{aut}{adx}{and}{Name Index}
```

Here, `aut` is the tag used to identify the author index, and "Name Index" is the title of the index. If the name of your main file is `root.tex`, then LaTeX will write the raw index entries to the file `root.adx`, and you will execute the following MakeIndex command to process the author index :

```
makeindex -o root.and root.adx
```

By default, the `\index` tags its argument with the page number (i.e., the value of `\thepage`), but occasionaly you may want to index items according to a different counter. For example, you may want an index that contains figure numbers instead of page numbers. To accomodate, this, the `\newindex` command takes an optional argument, which is the name of the command that generates the number that should be included in the index. For instance, to include the number of a figure, you might say

```
\newindex[thefigure]{fig}{fdx}{fnd}{Figures}
```

However, this introduces a new technicality : When creating an index with page numbers, the choice of which page number is to be written to the `aux` file should be deferred until the page containing the entry is shipped out to the `dvi` file, otherwise the wrong number will sometimes be chosen. However, when using counters other than the page counter, one normally wants the opposite behaviour : the number written to the `aux` file should be chosen immediately, otherwise every item on a given page will be tagged with the number of the last item on that page. So, when a counter is specified using the optional argument of `\newindex`, it is assumed that the counter should be evaluated immediately. If for some reason you need the choice to be deferred until the page is written to the `dvi` file, you can force this behaviour by putting a ∗ *after* the optional argument :

```
\newindex[thefigure]*{fig}{fdx}{fnd}{Figures}
```

(One consequence of this scheme is that if, for some reason, you need the choice of page number to be made immediately instead of being deferred until a page is shipped out to the `dvi` file, you can acomplish this by beginning your index declaration with

```
\newindex[thepage]*
```

`\renewindex`   The `\renewindex` command takes the same arguments as the `\newindex` command and can be used to redefine indexes that have been previously declared.

`\makeindex`   For backwards compatibility, the `\makeindex` command is redefined to use `\newindex`. It is essentially equivalent to

```
\newindex{default}{idx}{ind}{Index}
```

The index labeled `default` is special : it is the one that will be used by `\index` and `\printindex` unless another index is specified (see below).

`\printindex`   The `\printindex` command is modified by the addition of an optional argument, which is the tag of the index that should be printed.

`\index`   The `\index` command is modified in two ways. First, there is a ∗-variant of the command that, in addition to putting its argument into an index, also typesets it on the page. Second, `\index` now takes an optional argument to indicate which index the new entry should be added to. If given, the optional argument should be the identifying tag of a previously-defined index. If no such tag is supplied, the `default` index (such as that opened by `\makeindex` above) is used.

`\shortindexingon`   Perhaps the most dubious feature of `index.sty` is that it allows you to define `\shortindexingoff`   the characters `^` and `_` to be abbreviations for `\index*` and `\index` outside of math mode. These abbreviations are enabled by the `\shortindexingon` command and disabled by the `\shortindexingoff` command. The scope of both of these latter commands is local to the current group. (This might be useful, for example, if you wanted the abbreviations turned on throughout most of the documentation, but `shortindexingon`   turned off in one particular environment.) In addition, `shortindexingon` can be used as an environment if that seems appropriate. **Warning : This feature is deprecated and will disappear in a future release of this package.**

`\proofmodetrue`   As mentioned above, the `showidx` document-style option has been merged
`\proofmodefalse`
`\indexproofstyle`

3

into `index.sty`. It can be turned on with `\proofmodetrue` and turned off with `\proofmodefalse`. When it is turned on, all index entries [2] will be put in the margin of the page where they appear. By default, they appear in the typewriter font at `\footnotesize`, but the user can override this with the `\indexproofstyle` command; for example,

```
\indexproofstyle{\footnotesize\it}
```

will cause them to be put in italics instead.

`\disableindex`   There are some circumstances where it might be helpful to suppress the writing of a particular index. The `\disableindex` command is provided for this purpose. It takes one argument, a comma-separated list of tags of the indexes that should be disabled. This command should come *before* the declarations for the indexes that are being disabled [3]. One situation where the `\disableindex` command might be useful is if there are so many indexes that you are exhausting TeX's supply of output streams [4]. For example, suppose you have 10 indexes, but only 5 output streams available for indexes. Then you could add a `\disableindex` command to the top of your file to suppress the writing of all but 5 of the indexes. (Note that the index entries would still get written to the `aux` file; they just wouldn't get copied to the individual raw index files at the end of the run.) At the end of the run, you could then re-run your main file a couple of times with different indexes disabled until you had created all of the raw index files. This is somewhat clumsy, but safer than any alternative I've come up with [5].

4   Caveats

In order to implement this style file, it's been necessary to modify a number of LaTeX commands seemingly unrelated to indexing, namely, `\@starttoc`, `\raggedbottom`, `\flushbottom`, `\addcontents`, `\markboth`, and `\markright`. Naturally, this could cause incompatibilities between `index.sty` and any style files that either redefine these same commands or make specific assumptions about how they operate. See Section 6 for explanations of why these various commands needed modification.

The redefinition of `\@starttoc` is particularly bad, since it introduces an incompatibility with the AMS document classes. This will be addressed soon.

---

2. Well, most, at least. There are some circumstances under which the index entries won't show up in the proofs, although they will show up in the index.

3. This limits its usefulness somewhat, but since the output file for an index is opened when the index is declared, the damage has already been done. We could close the file, but we can't prevent a new output stream from being allocated and we can't keep the old file from being truncated.

4. TeX only has 16 output streams, which are allocated with the `\newwrite` command. The standard LaTeX styles use from 3 to 7 of these, which should leave room for up to 9 indexes. Of course, if you have extra output files, then there will be fewer output streams left for indexes.

5. A less clumsy (for the user, at least) solution would be to read the `aux` file multiple times at the end of the run, each time writing just one of the raw index files. The main disadvantage of this scheme at present is that it would require a modification of `\enddocument`.

Unfortunately, it's also been necessary to modify the `theindex` environment, so if you don't like the default LaTeX definition, you'll need copy the definition of `theindex` from this file and modify it appropriately.

In the current implementation, `index.sty` uses one output stream for each index. Since there are a limited number of output indexes, this means that there is a limit on the number of indexes you can have in a document. See the description of `\disableindex` for a fuller discussion of this problem and one way around it.

5   To do's

It might be nice if the `\index*` command parsed its argument so that, for example, instead of writing '`\index{sin@$\sin$}$\sin$`', one could write '`index*{sin@$\sin$}`'. However, this is fraught with numerous dangers, and I'm both too lazy and too cowardly to undertake it now.

It would be reasonable to add support for `\makeglossary` and similar things, if they were well-defined enough to decide what the general syntax for defining them should be.

The documentation should be carefully read, edited, and finished, especially since it's still based on the 2.09 version, even though a few substantial changes have been made for the LaTeX 2ε version.

For some truly outlandish ideas, see the file `TODO` in the distribution.

6   The code

As is customary, identify this as a LaTeX 2ε package.

```
1 ⟨∗style⟩
2 \NeedsTeXFormat{LaTeX2e}[1995/06/01]
3
4 \ProvidesPackage{index}[2004/01/20 v4.2beta Improved index support (dmj)]
```

`\disableindex` The `\disableindex` should come before the declarations of the indexes it refers to. (Question : If an index has been disabled, should it show up in index proofs ? Maybe there should be a separate command to disable index proofs on and index-by-index basis.)

```
5 \def\disableindex#1{%
6     \@for\@tempa:=#1\do{%
7         \@namedef{disable@\@tempa}{}%
8         \@ifundefined{tf@\@tempa}{}{%
9             \PackageWarningNoLine{index}{It's too late to disable
10                the '\@tempa' index;\MessageBreak
11                \jobname.\@tempa\space has already
12                been opened for output. You \MessageBreak
13                should put the \string\disableindex\space command
14                before\MessageBreak
15                the declaration of the '\@tempa' index}%
16         }%
```

```
17      }%
18 }
```

The `\newindex` and `\renewindex` commands are defined on analogy with the
`\[re]newcommand` macros. Each index is identified by a unique tag, which is spe-
cified in the first required argument of `\newindex`. Much of the information about
the index labeled $\langle tag \rangle$ is kept in the macro `\idx@`$\langle tag \rangle$, so we can check to see
if a particular index has already been defined by checking whether `\idx@`$\langle tag \rangle$ is
defined. `\newindex` and `\renewindex` both check to see if their first argument
is already associated with an index and then either issue an appropriate error
message or call `\def@index`.

The `\if@newindex` flag will be used to keep `\renewindex` from re-allocating
`\write` and `\toks` registers later. The `\if@tempswa` switch will be used to deter-
mine whether the `\writes` for this index should be done `\immediately` or not.

```
19 \newif\if@newindex
20
21 \def\newindex{%
22      \@tempswafalse
23      \@ifnextchar[{\@tempswatrue\x@newindex}{\x@newindex[\thepage]}%
24 }
25
26 \def\x@newindex[#1]{%
27      \@ifstar {\@tempswafalse\y@newindex{#1}}
28              {\y@newindex{#1}}%
29 }
30
31 \def\y@newindex#1#2{%
32      \@ifundefined{idx@#2}%
33          {\@newindextrue\def@index{#1}{#2}}%
34          {%
35              \@latexerr{Index type '\string#2' already defined}\@ehc
36              \expandafter\@gobble\@gobbletwo
37          }%
38 }
39
40 \def\renewindex{%
41      \@tempswafalse
42      \@ifnextchar[{\@tempswatrue\x@renewindex}{\x@renewindex[\thepage]}%
43 }
44
45 \def\x@renewindex[#1]{%
46      \@ifstar {\@tempswafalse\y@renewindex{#1}}
47              {\y@renewindex{#1}}%
48 }
49
50 \def\y@renewindex#1#2{%
51      \@ifundefined{idx@#2}%
52          {%
53              \@newindextrue
```

```
54              \@latexerr{Index type '\string#2' not defined}\@ehc
55          }%
56          {\@newindexfalse}%
57      \def@index{#1}{#2}%
58 }
```

\@preamblecmds   Neither \newindex, \renewindex, nor \disableindex should be used anywhere
except inside style files or in the preamble of a document, so we add them to the
\@preamblecmds list.

```
59 \@onlypreamble\newindex
60 \@onlypreamble\renewindex
61 \@onlypreamble\disableindex
```

\def@index   \def@index does most of the work. First, it picks up the first three arguments
of the \[re]newindex command and stores the second two in an appropriate
\idx@ macro. The title of the index is treated differently, however, since it is
potentially fragile in a particularly odd way. To prevent mishaps, it is stored in
a token register. In addition to stashing away the information about the index,
\def@index also opens an appropriate output file if we are writing auxiliary files
(i.e., unless \nofiles is in effect).

```
62 \def\def@index#1#2#3#4{%
63      \@namedef{idx@#2}{#3:#4:#1}%
64      \expandafter\let\csname if@immediate@#2\endcsname\if@tempswa
65      \if@filesw
66          \if@newindex
67              \expandafter\newtoks\csname idxtitle@#2\endcsname
68          \fi
69          \@ifundefined{disable@#2}{%
70              \if@newindex
71                  \expandafter\newwrite\csname tf@#2\endcsname
72              \else
73                  \immediate\closeout\@nameuse{tf@#2}%
74              \fi
75              \immediate\openout\@nameuse{tf@#2}\jobname.#3 %
76              \PackageInfo{index}{Writing index file \jobname.#3}%
77          }
78          {\PackageInfo{index}{Index '#2' disabled -- not opening
79                      \jobname.#3}}%
80      \fi
81      \expandafter\csname idxtitle@#2\endcsname
82 }
```

\@second   These are useful macros for retrieving the second and third field of an index
\@third   specification.

```
83 \def\@second#1:#2:#3\@nil{#2}
84
85 \def\@third#1:#2:#3\@nil{#3}
```

7

\@nearverbatim `\@nearverbatim\foo` is much like `\meaning\foo`, except that it suppresses the "`macro ->`" string produced when `\meaning` expands a macro. It is used by `\@wrindex` to produce an "almost verbatim" copy of their arguments. This method replaces the use of `\@sanitize` from latex.tex and allows indexing macros to be used in places (such as inside macro arguments) where the original `\index` command could not. Thanks to Donald Arseneau ⟨`asnd@erich.triumf.ca`⟩ for pointing out this trick to me. (For more information on this trick, see Dirty Trick #3 of the TEXbook, page 382).

As defined, nearverbatim only works on macros. It would be nice if it could work with other tokens, but it's more important that it work only by expansion, which means we can't put in tests to see what the next token is.

```
86 \def\@nearverbatim{\expandafter\strip@prefix\meaning}
```

Now we define the `\index` macro itself. The following definitions are adapted from `latex.tex` v2.09 ⟨25 March 1992⟩.

\makeindex First we redefine `\makeindex` to define the default index using `\newindex`. We use `\edef` to make sure that `\indexname` gets expanded here. Otherwise we'll get into an infinite loop later on when we try to redefine `\indexname` inside the `\theindex` environment.

Unfortunately, this means that if the user changes `\indexname` in the preamble, the index will come out with the wrong heading.

```
87 \edef\makeindex{%
88     \noexpand\newindex{default}{idx}{ind}{\indexname}%
89 }
```

\if@silentindex
\if@addtoindex
\if@proofmode We need three new flags. The first, `\if@silentindex`, indicates whether the entry should be typeset in running text, as well as written out to the index; this is used to implement the `\index*` command. The second, `\if@addtoindex`, indicates whether entries should be written to the index; this is used to disable the `\index` command inside of page headings and tables of contents. The third, `\ifproofmode`, indicates whether index entries should be put in the margin of the page for proofing purposes.

```
90 \newif\if@silentindex\@silentindextrue
91
92 \newif\if@addtoindex\@addtoindextrue
93
94 \newif\ifproofmode\proofmodefalse
```

\index
\p@index
\x@index `\index` will be made self-protecting (a la `\em`, etc.) so it can be used inside, for example, sectioning commands. Unfortunately, to really make `\index` robust, we have to redefine some of LATEX's commands for dealing with tables of contents and page headings. (See below.) ∗sigh∗

```
95 \def\index{\protect\p@index}
96
97 \def\p@index{%
98     \if@silentindex\@bsphack\fi
```

```
 99      \@ifstar{\@silentindexfalse\@xindex}{\@silentindextrue\@xindex}%
100 }
101
102 \def\@xindex{\@ifnextchar[{\@index}{\@index[default]}}
```

\@index
\@@index
\@wrindex

The following is much more complicated than it should have to be. First, note the check to see if \index is equal to \@gobble. This is so I don't have to redefine \@outputpage, which temporarily disables \label, \index, and \glossary by \let'ing them equal to \@gobble. (For this reason, we have to be very careful to make sure that \index has expanded to \p@index before it gets to \@outputpage.) Second, note that if \if@addtoindex is false, we don't complain about undefined index types. This is because if your page headings, for example, are being typeset in all uppercase, you might end up with something like \index[AUT]... instead of \index[aut]....

```
103 \def\@index[#1]{%
104     \ifx\index\@gobble
105         \@addtoindexfalse
106     \fi
107     \def\@tempf{%
108         \begingroup
109             \@sanitize
110             \@@index{#1}%
111     }%
112     \if@addtoindex
113         \@ifundefined{idx@#1}%
114             {%
115                 \def\@tempf{%
116                     \@latexerr{Index type '\string#1' undefined}%
117                     \@ehc
118                     \@silentindextrue
119                     \@gobble
120                 }%
121             }%
122             {}%
123     \fi
124     \@tempf
125 }
126
127 \def\@@index#1#2{%
128     \endgroup
129     \if@addtoindex
130         \if@filesw\@wrindex{#1}{#2}\fi
131         \ifproofmode\@showidx{#2}\fi
132     \fi
133     \if@silentindex
134         \expandafter\@esphack
135     \else
136         \@silentindextrue#2%
137     \fi
```

9

```
138 }
139
140 \def\@wrindex#1#2{%
141     \begingroup
142         \def\@tempa{#2}%
143         \edef\@tempb{\@nameuse{idx@#1}}%
144         \edef\@tempb{\expandafter\@third\@tempb\@nil}%
145         \csname if@immediate@#1\endcsname \else
146             \expandafter\let\csname\@tempb\endcsname\relax
147         \fi
148         \edef\@tempa{%
149             \write\@auxout{%
150                 \string\@writefile{#1}{%
151                     \string\indexentry{\@nearverbatim\@tempa}%
152                                     {\@nameuse{\@tempb}}%
153                 }%
154             }%
155         }%
156     \expandafter\endgroup\@tempa
157     \if@nobreak\ifvmode\nobreak\fi\fi
158 }
```

\seename The following are adapted from `makeidx.sty`, v2.09 ⟨21 Oct 91⟩. \index@prologue
\see adapted from `doc.dtx`. `theindex` based on version from `classes.dtx`, v1.3g, 26
\printindex June 1995.
\@printindex
```
159 \providecommand{\seename}{see}
160
161 \providecommand*{\see}[2]{\emph{\seename} #1}
162
163 \@ifclassloaded{article}{%
164
165     \renewenvironment{theindex}{%
166         \edef\indexname{\the\@nameuse{idxtitle@\@indextype}}%
167         \if@twocolumn
168             \@restonecolfalse
169         \else
170             \@restonecoltrue
171         \fi
172         \columnseprule \z@
173         \columnsep 35\p@
174         \twocolumn[%
175             \section*{\indexname}%
176             \ifx\index@prologue\@empty\else
177                 \index@prologue
178                 \bigskip
179             \fi
180         ]%
181         \@mkboth{\MakeUppercase\indexname}%
182                 {\MakeUppercase\indexname}%
183         \thispagestyle{plain}%
```

```latex
184          \parindent\z@
185          \parskip\z@ \@plus .3\p@\relax
186          \let\item\@idxitem
187     }{%
188          \if@restonecol
189               \onecolumn
190          \else
191               \clearpage
192          \fi
193     }
194 }{%
195     \renewenvironment{theindex}{%
196          \edef\indexname{\the\@nameuse{idxtitle@\@indextype}}%
197          \if@twocolumn
198               \@restonecolfalse
199          \else
200               \@restonecoltrue
201          \fi
202          \columnseprule \z@
203          \columnsep 35\p@
204          \twocolumn[%
205               \@makeschapterhead{\indexname}%
206               \ifx\index@prologue\@empty\else
207                    \index@prologue
208                    \bigskip
209               \fi
210          ]%
211          \@mkboth{\MakeUppercase\indexname}%
212                  {\MakeUppercase\indexname}%
213          \thispagestyle{plain}%
214          \parindent\z@
215          \parskip\z@ \@plus .3\p@\relax
216          \let\item\@idxitem
217     }{%
218          \if@restonecol
219               \onecolumn
220          \else
221               \clearpage
222          \fi
223     }
224 }

225
226 \def\printindex{\@ifnextchar[{\@printindex}{\@printindex[default]}}
227
228 \def\@printindex[#1]{%
229     \@ifnextchar[{\@print@index[#1]}{\@print@index[#1][]}%
230 }
231
232 \long\def\@print@index[#1][#2]{%
233     \def\@indextype{#1}%
```

```
234        \long\def\index@prologue{#2}%
235        \@ifundefined{idx@#1}%
236            {\@latexerr{Index type '\string#1' undefined}\@ehc}%
237            {%
238                \edef\@tempa{\@nameuse{idx@#1}}%
239                \edef\@tempa{%
240                    \noexpand\@input@{\jobname.\expandafter\@second\@tempa\@nil}%
241                }%
242                \@tempa
243            }%
244 }
```

\@indexstar@   Now we set things up for \shortindexing. [6] First, we define a one-token shorthand
                for \index*. This will be needed in the definition of \idx@activehat.

```
245 \def\@indexstar@{\index*}
```

\idx@activehat   Next, we define the values that ^ and _ will have when shortindexing is turned on.
\idx@activebar
```
246 \def\idx@activehat{%
247     \relax
248     \ifmmode\expandafter\sp\else\expandafter\@indexstar@\fi
249 }
250
251 \def\idx@activebar{%
252     \relax
253     \ifmmode\expandafter\sb\else\expandafter\index\fi
254 }
```

\shortindexingon   Now we define the \shortindexingon and \shortindexinoff commands to turn
\shortindexingoff   shortindexing on and off (surprise!). \shortindexingon saves the old definitions
                    and \catcode's of ^ and _ so they can later be restored by \shortindexingoff.
                    Both of these make their changes local to any enclosing group, so they can be
                    used as declarations to disable or enable shortindexing temporarily. In addition,
                    shortindexingon can also be used as an environment.
                        This is potentially very confusing. My basic rationale (if it can be described as
                    such) was that under normal circumstances, one would put \shortindexingon in
                    the preamble of one's document, and never want to turn it off. \shortindexingoff
                    is an attempt to make allowance for the contingency that someone might want to
                    turn shortindexing off, either permanently or temporarily.

```
255 \newif\if@shortindexing
256
257 \begingroup
258
259     \catcode'\^\active
260     \catcode'\_\active
261
262     \gdef\shortindexingon{%
```

---

6. **Warning :** This feature is deprecated and will be removed entirely in a future release of
this package.

```
263          \@shortindexingtrue
264          \chardef\old@idxhatcode\catcode'\^\relax
265          \chardef\old@idxbarcode\catcode'\_\relax
266          \catcode'\^\active
267          \catcode'\_\active
268          \let\old@idxhat ^%
269          \let\old@idxbar _%
270          \let^\idx@activehat
271          \let_\idx@activebar
272      }
273
274      \gdef\shortindexingoff{%
275          \if@shortindexing
276              \@shortindexingfalse
277              \let^\old@idxhat
278              \let_\old@idxbar
279              \catcode'\^\old@idxhatcode
280              \catcode'\_\old@idxbarcode
281          \fi
282      }
283
284 \endgroup
```

Now we take some code from `showidx.sty` and merge it into our new system. There are four reasons for redefining the commands here rather than just inputting `showidx.sty` (or requiring the user to do so). First, `showidx.sty` ends with a call to `\flushbottom`, which I want to avoid. Second, the instructions for successfully using `showidx.sty` along with `index.sty` would be somewhat tricky. This way, I can just tell users not to use `showidx.sty` at all. Third, I need to make some alterations to `\@showidx` anyway. In particular, (a) I need to add the `\@sanitizeat` command so this works correctly with AMS-LaTeX and (b) I want to add the `\indexproofstyle` command so the user can customize the size and font used for the index proofs. Finally, `showidx.sty` has at least two annoying bugs in it. See the edit-history for version 2.01 for a description.

`\@indexbox`   This code is adapted from `showidx.sty`, v2.09 ⟨16 Jun 1991⟩.

```
285 \newinsert\@indexbox
286
287 \dimen\@indexbox\maxdimen
```

`\@sanitizeat`   The definition of `\@sanitizeat` is slightly tricky, since we need `@` to be active when this macro is defined, but we also need it to be part of the control sequence name.

```
288 \begingroup
289      \catcode'\@\active
290      \expandafter\gdef\csname\string @sanitizeat\endcsname
291          {\def @{\char'\@}}
292 \endgroup
```

```
\indexproofstyle
        \@showidx   293 \newtoks\indexproofstyle
        \@leftidx   294
       \@rightidx   295 \indexproofstyle{\footnotesize\reset@font\ttfamily}
          \@mkidx   296
     \raggedbottom  297 \def\@showidx#1{%
      \flushbottom  298     \insert\@indexbox{%
         \@texttop  299         \@sanitizeat
                    300         \the\indexproofstyle
                    301         \hsize\marginparwidth
                    302         \hangindent\marginparsep \parindent\z@
                    303         \everypar{}\let\par\@@par \parfillskip\@flushglue
                    304         \lineskip\normallineskip
                    305         \baselineskip .8\normalbaselineskip\sloppy
                    306         \raggedright \leavevmode
                    307         \vrule \@height .7\normalbaselineskip \@width \z@\relax#1\relax
                    308         \vrule \@height\z@ \@depth.3\normalbaselineskip \@width\z@\relax
                    309     }%
                    310     \ifhmode\penalty\@M \hskip\z@skip\fi
                    311 }
                    312
                    313 \def\@leftidx{\hskip-\marginparsep \hskip-\marginparwidth}
                    314
                    315 \def\@rightidx{\hskip\columnwidth \hskip\marginparsep}
                    316
                    317 \def\@mkidx{%
                    318     \vbox to \z@{%
                    319         \rlap{%
                    320             \if@twocolumn
                    321                 \if@firstcolumn \@leftidx \else \@rightidx \fi
                    322             \else
                    323                 \if@twoside
                    324                     \ifodd\c@page \@rightidx \else \@leftidx \fi
                    325                 \else
                    326                     \@rightidx
                    327                 \fi
                    328             \fi
                    329             \box\@indexbox
                    330         }%
                    331         \vss
                    332     }%
                    333 }
                    334
                    335 \def\raggedbottom{%
                    336     \def\@textbottom{\vskip\z@ plus.0001fil}%
                    337     \let\@texttop\@mkidx
                    338 }
                    339
                    340 \def\flushbottom{\let\@textbottom\relax \let\@texttop\@mkidx}
```

```
341
342 \let\@texttop\@mkidx
```

Now, this next bit really gets up my nose. The only way to make sure that the \index command gets handled correctly when used inside of sectioning commands is to redefine a bunch of LaTeX's table of contents and running-heads macros. *blech* Fragility rears its ugly head again.

These are based on `latex.tex` 2.09 ⟨25 March 1992⟩.

\addcontentsline We need to redefine \addcontentsline to keep it from expanding \index commands too far. In particular, we have removed \index from the list of macros that are set equal to \@gobble and we substitute \@vwritefile for \@writefile. This latter change also means that we can simplify the definition of \protect somewhat.

```
343 \CheckCommand\addtocontents[2]{%
344    \protected@write\@auxout
345        {\let\label\@gobble \let\index\@gobble \let\glossary\@gobble}%
346        {\string\@writefile{#1}{#2}}%
347 }
348
349 \renewcommand{\addtocontents}[2]{%
350     \protected@write\@auxout
351        {\let\label\@gobble \let\glossary\@gobble}%
352        {\string\@writefile{#1}{#2}}%
353 }
```

\@starttoc We need to redefine \@starttoc to \@addtoindexfalse so that items don't get written to the index from within tables of contents. The only change here is the addition of \@addtoindexfalse.

Unfortunately, this will break pretty badly with the AMS document classes, since they redefine \@starttoc to take two arguments rather than one. This must be addressed.

```
354 \let\old@starttoc\@starttoc
355
356 \renewcommand{\@starttoc}[1]{%
357     \begingroup
358         \@addtoindexfalse
359         \old@starttoc{#1}%
360     \endgroup
361 }
```

\markboth Finally, we have to redefine \markboth and \markright to keep them from disa-
\markright bling the expansion of \index while putting section heads into the \mark. Otherwise, we'd end up with "\index" in the mark, which would cause problems when \@outputpage redefines \index to be equal to \@gobble. Instead, we want \index to expand to \p@index in the \mark, so we retain control over what happens in \@outputpage.

This time, the only change is to remove `\index` from the list of macros that are `\let` equal to `\relax`.

```
362 \CheckCommand*{\markboth}[2]{%
363   \begingroup
364     \let\label\relax \let\index\relax \let\glossary\relax
365     \unrestored@protected@xdef\@themark {{#1}{#2}}%
366     \@temptokena \expandafter{\@themark}%
367     \mark{\the\@temptokena}%
368   \endgroup
369   \if@nobreak\ifvmode\nobreak\fi\fi}
370 \CheckCommand*{\markright}[1]{%
371   \begingroup
372     \let\label\relax \let\index\relax \let\glossary\relax
373     \expandafter\@markright\@themark {#1}%
374     \@temptokena \expandafter{\@themark}%
375     \mark{\the\@temptokena}%
376   \endgroup
377   \if@nobreak\ifvmode\nobreak\fi\fi}
378
379 \renewcommand{\markboth}[2]{%
380   \begingroup
381     \let\label\relax \let\glossary\relax
382     \unrestored@protected@xdef\@themark {{#1}{#2}}%
383     \@temptokena \expandafter{\@themark}%
384     \mark{\the\@temptokena}%
385   \endgroup
386   \if@nobreak\ifvmode\nobreak\fi\fi}
387
388 \renewcommand{\markright}[1]{%
389   \begingroup
390     \let\label\relax \let\glossary\relax
391     \expandafter\@markright\@themark {#1}%
392     \@temptokena \expandafter{\@themark}%
393     \mark{\the\@temptokena}%
394   \endgroup
395   \if@nobreak\ifvmode\nobreak\fi\fi}
396 ⟨/style⟩
```

7   Edit history

v1.00 (4 Mar 1993) initial version, posted to comp.text.tex.

v1.01 (4 Mar 1993) added `\renewindex` command and checking to make sure index is (or is not) defined in `\newindex`, `\index` and `\printindex`. Also tightened up the code in various places and added check to make sure file is only loaded once.

v2.00 (24 Mar 1993) added support for `\index*`, proofmode, `\shortindexingon` and `\shortindexingoff`.

v2.01 (24 Jun 1993) Fixed 3 bugs. (1) If proofmode was turned on, then something like "`\indexWORDWORD`" would suppress the hyphenation of WORD. This was fixed by adding "`\penalty\@M\hskip\z@skip`" to the end of `\@showidx`. (This is just the definition of `\allowhyphens` borrowed from `german.sty`, v2 ⟨4 Nov 1988⟩). (2) The `\hbox` in `\@mkidx` was being set at its natural width, which had a tendency to interfere with the width of the page. The `\hbox` is now replaced by `\rlap`. (3) If the title of an index (i.e., the fourth argument of `\newindex`) contained a particularly fragile command like `\d`, havoc would ensue when `\theindex` tried to extract the title. Titles are now kept in token registers to prevent such unpleasantness. Bugs (2) and (3) were reported by Dominik Wujastyk ⟨`D.Wujastyk@ucl.ac.uk`⟩ on 24 June 1993. Note that bugs (1) and (2) are actually bugs in showidx.sty, v2.09 ⟨16 Jun 1991⟩.

v2.02 (25 Jun 1993) Rewrote the code that implements the short indexing commands (`^` and `_`) to make index.sty compatible with other style files that need to make `^` and `^` active in some contexts. See the code for more details.

v2.03 (30 Jun 1993) Once again rewrote the code that implements the short indexing commands. Dumped the shortindexing environment and rewrote the `\shortindexingon` and `\shortindxingoff` commands to save and restore the `\catcode`'s and meanings of `^` and `^` in the safest possible (I hope) order. Also added the `\if@shortindexing` flag to keep `\shortindexingoff` from doing anything if it is called outside of the scope of a `\shortindexingon` command. (Question : Should `\shortindexingon` check that flag before doing anything ?)

v2.04 (beta) (14 Jul 1993) Added `\disableindex` command. Added `\newindex` and `\renewindex` to `\@preamblecmds`. Added `\if@newindex` flag to `\@newindex` to prevent `\renewindex` from re-allocating new `\write` and `\toks` registers. Rewrote using `doc.sty` and DocStrip. Also cleaned up the code somewhat.

v3.00 (15 Jul 1993) Made further minor tweaks to code and internal documentation. Booted version number up to 3.00 and released on the world.

v3.01 (19 Jul 1993) Fixed DocStrip CheckSum.

v3.02 (15 Sep 1993) Corrected spelling of `\@shortindexingfalse` in definition of `\shortindexingoff`. Thanks to Hendrik G. Seliger ⟨`hank@Blimp.automat.uni-essen.de`⟩ for this bug report. Also added redefinitions of `\@leftmark` and `\@rightmark` to fix a bug reported by Dominik Wujastyk ⟨`D.Wujastyk@ucl.ac.uk`⟩.

v3.03 (beta) (20 Feb 1994) Added `\long` to the definition of `\@ifundefined` to cover the unlikely contingency that someone wanted to use, for example, `\string\par` in the middle of a control sequence name. Added an optional argument to `\newindex` to specify which counter to use in place of `\thepage`. The first change was suggested by Martin Schröder ⟨`l15d@zfn.uni-bremen.de`⟩ ; the second was suggested independently by Schröder and Stefan Heinrich Höning ⟨`hoening@pool.informatik.rwth-aachen.de`⟩.

The `\@newindex` command was renamed `\def@index`. Also fixed the `\disableindex` command.

v3.04 (7 Mar 1994) Rewrote the user documentation (Sections 1–5) and released on the world. Also deleted some extraneous spaces that had crept into some macros.

v4.00beta, (20 Feb 1995) Preliminary conversion to a native LaTeX 2_ε package. Fixed `\@printindex` to work under LaTeX 2_ε (bug reported by Carsten Folkertsma ⟨`cai@butler.fee.uva.nl`⟩). Removed much code that had been put in to work around various ancient versions of LaTeX 2.09. Added `\index@prologue` support (modelled on `doc.sty`) at suggestion of Nick Higham ⟨`higham@ma.man.ac.uk`⟩.

v4.01beta (28 Sep 1995) Rewrote as a LaTeX 2_ε package (finally !). Changes too numerous to list, but in general deleted some now-superfluous code, replaced some tricks by tricks from the LaTeX 2_ε kernel, and added some bullet-proofing. Much still remains to be done, but this should be good enough for testing.

Changed definition of `\protect` in `\markright` and `\markboth` to fix bug reported by Dominik Wujastyk.

? ? ? (5 Jan 2004)

8 The sample file

```
397 ⟨∗sample⟩
398 %% latex sample.tex
399 %% makeindex sample
400 %% makeindex -o sample.and sample.adx
401 %% makeindex -o sample.nnd sample.ndx
402 %% makeindex -o sample.lnd sample.ldx
403 %% latex sample.tex
404
405 \documentclass{book}
406 \usepackage{index}
407
408 \listfiles
409
410 \makeindex
411 \newindex{aut}{adx}{and}{Name Index}
412 \newindex{not}{ndx}{nnd}{List of Notation}
413
414 \newindex[theenumi]{list}{ldx}{lnd}{Items}
415
416 \shortindexingon
417
418 \proofmodetrue
419
420 \def\aindex{\index*[aut]}
```

```
421
422 \begin{document}
423
424 \tableofcontents
425
426 \newpage
427
428 \chapter{Here is a ^[aut]{chapter} title}
429
430 \section{Section header\index[aut]{section}}
431
432 Here is some text.\index{subject}
433
434 Here is \index[not]{notation}some more \index[not]{sin@$\sin$}
435 text.
436
437 \newpage
438
439 Here is some ^{more} _[not]{notation} text.
440
441 Here is yet more \aindex{text}.
442
443 \section{Another Section header _[aut]{section2}}
444
445 And here is some math: $x^1_b$.
446
447 Here is an ^[aut]{index} entry \fbox{inside an
448 \index[not]{min@$\min$}fbox}
449
450 \fbox{Here is an ^[aut]{entry} in a box.}
451
452 \section{An indexed list environment}
453
454 \begin{enumerate}
455
456 \item
457 First item
458
459 \item
460 Second item\index[list]{second item}
461
462 \item
463 Third item
464
465 \newpage
466
467 \item
468 Fourth item
469
470 \item
```

19

```
471 Fifth item\index[list]{fifth item}
472
473 \item
474 Sixth item
475
476 \end{enumerate}
477
478 \printindex[not]
479
480 \printindex[aut][Here is a prologue for the author index.
481 Note that it is set in a single column at the top of the
482 first page of the index.]
483
484 \printindex[list]
485
486 \printindex
487
488 \end{document}
489 ⟨/sample⟩
```

Index

Les numéros en italique renvoient à la page où se trouve l'entrée correspondante; les numéros soulignés renvoient à la ligne de code de la définition; les numéros en romain renvoient aux lignes de code où l'entrée est utilisée.

20