Standard LaTeX 2ε packages makeidx and showidx

Johannes Braams   David Carlisle   Alan Jeffrey
Leslie Lamport   Frank Mittelbach   Chris Rowley
Rainer Schöpf

1994/11/11

1   Description

1.1   Makeidx

The package `makeidx` provides two new commands, `\see` and `\printindex`.

`\see`   The command `\see` can used in the index to cross reference to other items.

`\printindex`   This command can be used to include the sorted and formatted index in the document.

1.2   Showidx

The package `showidx` changes a number of internal LaTeX 2ε commands in such a way that each index entry is shown in the margin on the page where the entry appears. This package was originally meant to be used with the `report` and `book` document classes, but works with other classes as well. It makes `\flushbottom` the default.

2   The DOCSTRIP modules

The following modules are used in the implementation to direct DOCSTRIP in generating the external files:

makeidx	produce makeidx.sty
showidx	produce showidx.sty
driver	produce a documentation driver file

3   The documentation driver file

The next bit of code contains the documentation driver file for TeX, i.e., the file that will produce the documentation you are currently reading. It can be extracted from this file by the DOCSTRIP program.

```
1 ⟨∗driver⟩
2 \documentclass{ltxdoc}
3 \begin{document}
4 \DocInput{makeindx.dtx}
5 \end{document}
6 ⟨/driver⟩
```

4 Implementation

4.1 Identification

Announce the package and its version:

```
7 ⟨makeidx⟩\ProvidesPackage{makeidx}
8 ⟨showidx⟩\ProvidesPackage{showidx}
9              [2000/03/29 v1.0m Standard LaTeX package]
```

4.2 Makeidx

\see This macro discards its second argument (typically a page number) and just prints \seename together with the entry the reader is pointed to.

```
10 ⟨*makeidx⟩
11 \newcommand*\see[2]{\emph{\seename} #1}
```

\seealso This macro discards its second argument (typically a page number) and just prints \alsoname together with the entry the reader is pointed to. We use \providecommand to retain compatibility with existing files that define this macro.

```
12 \providecommand*\seealso[2]{\emph{\alsoname} #1}
```

\printindex This command simply inputs the (formatted) index if it exists, otherwise a warning is issued.

```
13 \newcommand\printindex{\@input@{\jobname.ind}}
```

\seename This package is for documents prepared in the English language. To prepare a version for another language, various English words must be replaced. All the English words that require replacement are defined below in command names.

```
14 \providecommand\seename{see}
```

We used \providecommand in case the command is already defined by a package loaded earlier.

\alsoname This macro discards its second argument (typically a page number) and just prints \alsoname together with the entry the reader is pointed to. We use \providecommand to retain compatibility with existing files that define this macro.

```
15 \providecommand*\alsoname{see also}
16 ⟨/makeidx⟩
```

4.3 showidx

\indexbox This package uses TeX's insert mechanism, therefore it needs to allocate an insert register.

```
17 ⟨*showidx⟩
18 \newinsert\indexbox
19 \dimen\indexbox=\maxdimen
```

\index This is a modified default definition for the user level \index command. The difference is the change of the category code of the space character.

```
20 \renewcommand\index{\@bsphack\begingroup
21                    \@sanitize\catcode32=10\relax\@index}
```

\makeindex The same change has to be included in the user level \makeindex command, which changes the definition of \index to actually write index entries to an external file.

```
22 \renewcommand\makeindex{\if@filesw \newwrite\@indexfile
23   \immediate\openout\@indexfile=\jobname.idx
24   \def\index{\@bsphack\begingroup
25     \def\protect####1{\string####1\space}\@sanitize
26     \catcode32=10 \@wrindex\@indexfile}\typeout
27   {Writing index file \jobname.idx }\fi}
```

\@wrindex This macro takes care of writing the index entries to a file. The definition is modified to call \@showidx.

```
28 \def\@wrindex#1#2{\let\thepage\relax
29    \xdef\@gtempa{\write#1{\string
30        \indexentry{#2}{\thepage}}}\endgroup\@gtempa
31    \@showidx{#2}\if@nobreak \ifvmode\nobreak\fi\fi\@esphack}
```

\@index When the user didn't use the \makeindex command, the \index macro calls \@index, which normally does basically nothing. This package changes the definition to call \@showidx, which includes the entry in the list of indexentries on the current page.

```
32 \def\@index#1{\@showidx{#1}\endgroup\@esphack}
```

\@showidx This macro adds the current index entry to the insert \indexbox. The \indexbox is typeset as a flushleft paragraph.

```
33 \def\@showidx#1{%
34    \insert\indexbox{\small
35        \hsize\marginparwidth
36        \hangindent\marginparsep \parindent\z@
37        \everypar{}\let\par\@@par \parfillskip\@flushglue
38        \lineskip\normallineskip
39        \baselineskip .8\normalbaselineskip\sloppy
40        \raggedright \leavevmode
41        \vrule \@height .7\normalbaselineskip \@width \z@\relax
42            #1\relax
43        \vrule \@height \z@ \@depth .3\normalbaselineskip \@width \z@}}
```

\raggedbottom
\flushbottom The definition of these macros from `latex.dtx` is changed here to add the execution of \@mkidx to \@texttop, which is executed at the top of each page.

```
44 \renewcommand\raggedbottom{\def\@textbottom{\vskip
45        \z@ plus.0001fil}\let\@texttop\@mkidx}
46 \renewcommand\flushbottom{\let\@textbottom\relax
47                          \let\@texttop\@mkidx}
```

\@mkidx This macro actually typesets the box containing all the index entries on the current page. They will occur on the left or the right side of the text, or both, depending on the setting of the switches \if@twocolumn and \if@twoside.

```
48 \def\@mkidx{\vbox to \z@{\hbox{\if@twocolumn
49    \if@firstcolumn \@leftidx \else \@rightidx \fi
50 \else \if@twoside \ifodd\c@page \@rightidx
51                  \else \@leftidx \fi
52        \else \@rightidx \fi
53    \fi
54    \box\indexbox}\vss}}
```

\@leftidx
\@rightidx These macros give the amount of displacemant for the \indexbox.

```
55 \def\@leftidx{\hskip-\marginparsep \hskip-\marginparwidth}
56 \def\@rightidx{\hskip\columnwidth \hskip\marginparsep}
```

To make this work we have to execute either \raggedbottom or \flushbottom. Assuming this package is used most often with the document classes `report` and `book`, we execute \flushbottom.

```
57 \flushbottom
58 ⟨/showidx⟩
```