The listofsymbols.sty package (v0.2)

Stefan Spreng
August 2003

Abstract

listofsymbols provides commands to (a) automatically create a list of symbols (also called notation or nomenclature) and (b) handle symbols logically, i.e. use a command that is expanded to the desired output rather than ‘hardcoding’ the output into the text.

This helps to ensure consistency throughout the text, especially if there is a chance that symbols will be changed at some stage. Additionally, you can keep all definitions of symbols in a separate file.

This package is more or less a combination of what the packages nomencl.sty and formula.sty do. The concept of creating the list of symbols, though, is different from the way nomencl.sty does it.

Contents

1 User Interface 2
  1.1 Options ................................................. 2
  1.2 Macros ................................................... 2

2 Examples 7
  2.1 Example 1 ............................................... 7
  2.2 Example 2 ............................................... 8
  2.3 Example 3 ............................................... 8

3 Contact 8

4 The Code 9
1 User Interface

1.1 Options

draft  Default.

final  Removes the macronames from the lists. Symbols that are not used in the document are omitted from the List of Symbols and from the List of Subscripts.

Final  Similar to final. The difference is that the .sym and .sub files are not changed any more. Use this mode when your document is ready and you have sorted the .sym and .sub files manually.

nomencl  Typesetting the list of symbols with the package nomencl (symbols and subscripts are in one list). With this option, the macros described in this documentation call the appropriate commands that nomencl.sty provides. See the documentation of nomencl.sty for details of the layout.

nopageno  Default.

pageno  Inserts the number of the page on which a symbol or subscript is defined.

usexspace  Default. Uses the package xspace to insert an ‘intelligent’ space after the commands.

noxspace  Do not load package xspace. In this case a command must be followed by a backslash and a space if you want a space in the output (This is the LaTeX standard).

You can use only one of the options draft, final, Final or nomencl and only one of the options nopageno or pageno.

1.2 Macros

\opensymdef  All \newsym and \newsub commands must be between the commands \opensymdef and \closesymdef. A \listofsymbols or a \listofsubscripts must be outside the region enclosed by these commands. Otherwise you will get errors. See section 2 for examples.

\newsym  The macro \newsym assigns the desired output of a symbol or variable to a macro which can thereafter be used like any other macro. \newsym takes one optional argument and two mandatory arguments.
\newsym[ description ]{ macroname }{ output }

The optional argument is the description that will appear in the list of symbols. The first mandatory argument is the name of the macro and the second mandatory argument is the desired output. Note that the definition of \newsym includes output in a \ensuremath{} command. If there is no description then the symbol is included in the list of symbols in draft mode, but not in final mode.

Examples can be found in section 2. You will probably notice that all the macros start with sym. That’s because I think that this makes it easier to distinguish between symbols and other macros you define in a document. Personally, I use a y and an s as the first characters to indicate that a command is a symbol or a subscript, it’s shorter . . .

The description of a macro can be accessed with doc appended to the macroname. A string that can be used inside a tabular environment can be obtained by appending tabdoc to the command. If tabdoc is appended, then the macro expands to output & description

Example:

\newsym[Energy]{symE}{E}
The symbol \symE means \symEdoc

\begin{tabular}{ll}
Symbol&Description\\
\symEtabdoc\\
\end{tabular}

\newsym The macro \newsym creates a subscript much in the same way as \newsym creates a symbol. The syntax is

\newsym[ description ]{ subscription }{ output }

\newsym The macro \subsep separates two subscripts and thus avoids a LaTeX error. Its syntax is

\subsep[ separator ]

By default, the separator is empty, i.e. the second subscript simply follows the first one.

If you want to use a subscript after a symbol that does not have a subscript yet, simply put if after the symbol, e.g. \symx\suby. If the symbol already
has a subscript, you have to put a $\text{\textbackslash subsep}$ in front of the subscript. In regular text, you should enclose such a construct with $\text{'s}$ to avoid space between the symbol and the subscript.

Example:

\newsym{symx}{x}
\newsym{syma}{a\_b}
\newsymb{suby}{y}
\newsymb{subz}{z}

Usage: $\text{symx}\text{\textbackslash suby}$
Or: $\text{\textbackslash symx}\text{\textbackslash suby}\text{\textbackslash subsep}\text{\textbackslash subz}$
Finally: $\text{\textbackslash symx}\text{\textbackslash suby}\text{\textbackslash subsep[\textbackslash,]}\text{\textbackslash subz}$

In an equation:
\[ \text{\textbackslash symx}\text{\textbackslash suby} = \text{\textbackslash syma}\text{\textbackslash subsep[\textbackslash,]}\text{\textbackslash suby} \]

\textbf{\textbackslash listofsymbols} The command \textbf{\textbackslash listofsymbols} generates a list of the symbols, that were created with \textbf{\textbackslash newsym}. The symbols are not sorted. You have to do that manually by sorting the lines in the .sym and .sub files, for example with an editor or a spreadsheet. Once you have sorted the symbols and do not want to have the files changed any more, use the \textbf{Final} mode. Before using the \textbf{Final} mode, you must compile the document at least once in \textbf{final} mode to get the proper .sym and .sub files.

A typical sequence would be

- Compile in \textbf{draft} mode (as often as you want)
- Compile in \textbf{final} mode (at least once)
- Sort and edit the .sym and .sub files
- Compile in \textbf{Final} mode (as often as you want). If you add new symbol or subscript definitions now, they will not appear in the list of symbols or subscripts. If you use \textbf{draft} or \textbf{final} mode now, the edited version of the .sym and .sub files will be overwritten.

Note that the command \textbackslash listofsymbols \textbf{must} be outside the region that is enclosed by the \textbackslash opensymdef and \textbackslash closessymdef commands.

In \textbf{draft} mode, which is the default, the names of the macros are included in the lists. That makes it easier to keep track of the macro names and the corresponding output. The number of times the symbol was used in the document is given in parentheses. Symbols that do not have a \emph{description} are included in the list as well.
In final mode, the macro-names disappear. Symbols without a description (or an empty description) and symbols that are not used in the document are not included in the list. This allows you to keep all the symbols you normally need in a separate file that contains only \newsym and \newsub commands. This file can be included in the main file, e.g. with \include or \input. See the .log file for information which symbols have been omitted from the lists.

The Final mode is similar to final. The difference is that the .sym and .sub files are not changed. Use this mode when your document is ready and you have sorted the .sym and .sub files manually. The first pair of braces after the \printsymline in a line of the .sym and .sub files is not used by listofsymbols. You can use it for example to help the sorting process.

Example: It is valid to change the line
\printsymline{\ell}{\ell}{syml}{Length}{1}
manually into
\printsymline{l}{\ell}{syml}{Length}{1}

In nomencl mode the glossary has to be generated manually, for example by entering
\makeindex filename.glo -s nomencl.ist -o filename.gls
at the command line. Read the documentation of the nomencl package for more information.

\symwidth The length \symwidth is the space reserved for the symbol on the left side of each line and is by default set to 2.5 cm. If you have long symbols you may have to change that, for example with
\setlength{\symwidth}{3cm}.

\listofsubscripts Similar to \listofsymbols, but for the subscripts obviously.

\listofboth Creates both a list of symbols and a list of subscripts with the heading ‘Notation’ above them.

\symheadingname \subheadingname The headings of the lists are stored in \symheadingname and \subheadingname. In order to change it you can use for example
\renewcommand{\symheadingname}{ New Heading }

\markasused \markasunused If you want to decide yourself whether a symbol or subscript should be included in the lists, you can issue a \markasused or \markasunused command. A \markasunused command should occur after the last call of the macro.

Note that \markasused works only if the description of a symbol or subscript is not empty (why would one want to have a symbol without a description in the list of symbols?). If, for some reason, you want such a
symbol to be in the list of symbols, change its description in the `\newsym` command to something invisible, e.g. a space.

Syntax:

```latex
\markasused { macroname }
\markasunused { macroname }
```

Note that `macroname` must be given without the backslash.

\dontmarkasused If you want to get the output of a symbol without changing the “used”-flag, you can use the command `\dontmarkasused`. This is for example useful if you want to use a symbol in the description of another symbol. In this case you have to insert a `\noexpand` command before the macro `\dontmarkasused`.

Syntax:

```latex
\dontmarkasused { macroname }
```

Note that `macroname` must be given without the backslash.

Example:

```latex
\opensymdef
  \newsym{syma}{a}
  \newsym[Derivative of \noexpand\dontmarkasused{syma}]{symda}{\syma '}
\closesymdef
```

\losstring If the output of a symbol or subscript contains macros and they are not at the very beginning of the definition, then you have to insert a `\losstring` command in front of the macro.

Example:

```latex
\opensymdef
  \newsym{symA}{a}
  \newsym{symB}{\overline{\losstring\syma}}
  \newsym{symC}{\overline{\overline{\losstring\overline{\losstring\syma}}}}
\closesymdef
```
2 Examples

These examples are supposed to illustrate the implications of \texttt{opensymdef} and \texttt{closesymdef}.

2.1 Example 1

Here, the definitions are in the preamble

\begin{verbatim}
\documentclass{article}
\usepackage{listofsymbols}
\opensymdef
  \newsym[Energy]{symE}{E}
  \newsym[Mass]{symm}{m}
  \newsym[Speed of light]{symc}{c}
\closesymdef
\begin{document}
\[
\symE = \symm \symc^2
\]
where \symE is the energy \ldots
\listofsymbols
\end{document}
\end{verbatim}

Output: \begin{verbatim}
E = m \, c^2
\end{verbatim}

where $E$ is the energy \ldots

List of Symbols (draft)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E$</td>
<td>$\symE$ – Energy (yes)</td>
</tr>
<tr>
<td>$m$</td>
<td>$\symm$ – Mass (yes)</td>
</tr>
<tr>
<td>$c$</td>
<td>$\symc$ – Speed of light (yes)</td>
</tr>
</tbody>
</table>
2.2 Example 2

Here, the list of symbols is at the end of the document and the definitions are in the body.

\documentclass{article}
\usepackage{listofsymbols}

\begin{document}
\opensymdef
\newsym{[Energy]}{E} \newsym{[Mass]}{m} \newsym{[Speed of light]}{c}

\[\text{symE} = \text{symm} \cdot \text{symc}^2\]

where \text{symE} is the energy \ldots

\closesymdef
\listofsymbols
\end{document}

2.3 Example 3

Now, the list of symbols is before the definitions.

\documentclass{article}
\usepackage{listofsymbols}

\begin{document}
\listofsymbols
\opensymdef
\newsym{[Energy]}{E} \newsym{[Mass]}{m} \newsym{[Speed of light]}{c}

\[\text{symE} = \text{symm} \cdot \text{symc}^2\]

where \text{symE} is the energy \ldots

\closesymdef
\end{document}

3 Contact

If you have suggestions how this package can be improved, let me know:
4 The Code

1 \NeedsTeXFormat{LaTeX2e} \ProvidesPackage{listofsymbols}
2 \RequirePackage{ifthen} \RequirePackage{calc} \newboolean{b@nomencl}
3 \newboolean{b@final} \newboolean{b@Final} \newboolean{b@pageno}
4 \newboolean{b@xspace}
5 \DeclareOption{nomencl}{\setboolean{b@nomencl}{true}}
6 \DeclareOption{draft}{\setboolean{b@nomencl}{false}\setboolean{b@final}{false}\setboolean{b@Final}{false}}
7 \DeclareOption{final}{\setboolean{b@nomencl}{false}\setboolean{b@final}{true}\setboolean{b@Final}{false}}
8 \DeclareOption{Final}{\setboolean{b@nomencl}{false}\setboolean{b@final}{true}\setboolean{b@Final}{true}}
9 \DeclareOption{pageno}{\setboolean{b@pageno}{true}}
10 \DeclareOption{nopageno}{\setboolean{b@pageno}{false}}
11 \DeclareOption{usexspace}{\setboolean{b@xspace}{true}}
12 \DeclareOption{noxspace}{\setboolean{b@xspace}{false}}
13 \ExecuteOptions{draft,nopageno,usexspace}
14 \ProcessOptions
15
16 \newlength{\symindent}
17 \setlength{\symindent}{1.5em}
18 \newlength{\symwidth}
19 \setlength{\symwidth}{2.5cm}
20 \newlength{\sympagenowidth}
21 \ifthenelse{\boolean{b@nomencl}}{\RequirePackage{nomencl}}{}
22 \ifthenelse{\boolean{b@xspace}}{\RequirePackage{xspace}\newcommand{\spaceaftersym}{\xspace}}{\newcommand{\spaceaftersym}{} }
23 \ifthenelse{\boolean{b@pageno}}{\settowidth{\sympagenowidth}{9999}}{\setlength{\sympagenowidth}{0cm}}
24 \newcommand{\printsymline}[5]{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}
25 \ifthenelse{\boolean{b@final}}{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}}{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}}
26 \%#1: sortkey
27 \%#2: symbol
28 \%#3: macroname
29 \%#4: description
30 \%#5: page number
31 \newcommand{\printsymline}[5]{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}}
32 \ifthenelse{\boolean{b@final}}{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}}{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}}
33 \%#1: sortkey
34 \%#2: symbol
35 \%#3: macroname
36 \%#4: description
37 \%#5: page number
38 \newcommand{\printsymline}[5]{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}}
39 \ifthenelse{\boolean{b@final}}{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}}{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}}
\ifthenelse{\boolean{b@nomencl}}{
ewcommand{\newsym}[3][\] \@createsym{#1}{#2}{#3} \ifthenelse{\equal{#1}{}}{}{\nomenclature{\ensuremath{#3}}{#1}}}{\newcommand{\newsym}[3][\] \@createsym{#1}{#2}{#3} \addsymline{#3}{#3}{#2}{#1}{\@sym}}

\%#1: description
\%#2: macroname
\%#3: symbol
\newcommand{\@createsub}[3][\] \expandafter{\newcommand}{\csname#2\endcsname}{\relax\ensuremath{_{#3}}\spaceaftersym{}} \expandafter{\protected@xdef}{\csname#2isused\endcsname}{\yes}} \expandafter{\newcommand}{\csname#2doc\endcsname}{#1} \expandafter{\newcommand}{\csname#2tabdoc\endcsname}{\ensuremath{#3} & #1} \expandafter{\protected@xdef}{\csname#2isused\endcsname}{\no}}

\ifthenelse{\boolean{b@nomencl}}{\newcommand{\newsub}[3][\] \@createsub{#1}{#2}{#3} \ifthenelse{\equal{#1}{}}{}{\nomenclature{\ensuremath{#3}}{#1}}}{\newcommand{\newsub}[3][\] \@createsub{#1}{#2}{#3} \addsymline{#3}{#3}{#2}{#1}{\@sub}}

\newcommand{\subsep}[1][\] {\ensuremath{{}_{#1}{}}}

\newcommand{\symheadingname}{List of Symbols}
\newcommand{\subheadingname}{List of Subscripts}
\newcommand{\bothheadingname}{Notation}
\ifthenelse{\boolean{b@final}}{\newcommand{\symheading{}}{\section*{\symheadingname}} \newcommand{\subheading{}}{\section*{\subheadingname}} \ifthenelse{\boolean{b@pageno}}{\hfill{\bf Defined on page}}{}}

\ifthenelse{\boolean{b@nomencl}}{\newcommand{\newsymbol}[3][\] \@createsymbol{#1}{#2}{#3} \ifthenelse{\equal{#1}{}}{}{\nomenclature{\ensuremath{#3}}{#1}}}{\newcommand{\newsymbol}[3][\] \@createsymbol{#1}{#2}{#3} \addsymline{#3}{#3}{#2}{#1}{\@sym}}

\newcommand{\subsep}[1][\] {\ensuremath{{}_{#1}{}}}

\newcommand{\symheadingname}{List of Symbols}
\newcommand{\subheadingname}{List of Subscripts}
\newcommand{\bothheadingname}{Notation}
\ifthenelse{\boolean{b@final}}{\newcommand{\symheading{}}{\section*{\symheadingname}} \newcommand{\subheading{}}{\section*{\subheadingname}} \ifthenelse{\boolean{b@pageno}}{\hfill{\bf Defined on page}}{}}
\newcommand{\markasused}[1]{\expandafter\protected@xdef\csname#1isused\endcsname {yes}}
\newcommand{\los@temp}{}
\newcommand{\dontmarkasused}[1]{\protected@xdef\los@temp{\csname#1isused\endcsname}\csname#1\endcsname\%
\expandafter\protected@xdef\csname#1isused\endcsname{\los@temp}}
\AtEndDocument{
\renewcommand{\printsymline}[5]{\immediate\write\@syc{\string\newcommand{\expandafter\string\csname #3include\endcsname}{\csname #3isused\endcsname}}}\newwrite\@syc \immediate\openout\@syc=\jobname.syc
\IfFileExists{\jobname.sym}\@input@{\jobname.sym}{}\immediate\closeout\@syc
\renewcommand{\printsymline}[5]{\immediate\write\@suc{\string\newcommand{\expandafter\string\csname #3include\endcsname}{\csname #3isused\endcsname}}}\newwrite\@suc \immediate\openout\@suc=\jobname.suc
\IfFileExists{\jobname.sub}\@input@{\jobname.sub}{}\immediate\closeout\@suc}
\endinput