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

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 by 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` `\closeymdef` 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}
```

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

\newsub[*description*]{ *subscriptname* }{ *output* }

`\subsep` 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 `\subsep` in front of the subscript. In regular text, you should enclose such a construct with `$`'s to avoid space between the symbol and the subscript.

Example:

```
\newsym{symx}{x}
\newsym{syma}{a_b}
\newsub{suby}{y}
\newsub{subz}{z}

Usage: $\symx\suby$
Or: $\symx\suby\subsep\subz$
Finally: $\symx\suby\subsep[,]\subz$

In an equation:
\[ \symx\suby = \syma\subsep[,]\suby \]
```

`\listofsymbols` The command `\listofsymbols` generates a list of the symbols, that were created with `\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 `Final` mode. Before using the `Final` mode, you must compile the document at least once in `final` mode to get the proper `.sym` and `.sub` files.

A typical sequence would be

- Compile in `draft` mode (as often as you want)
- Compile in `final` mode (at least once)
- Sort and edit the `.sym` and `.sub` files
- Compile in `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 `draft` or `final` mode now, the edited version of the `.sym` and `.sub` files will be overwritten.

Note that the command `\listofsymbols` **must** be outside the region that is enclosed by the `\opensymdef` and `\closesymdef` commands.

In `draft` mode, which is the default, the names of the macros are inlcuded in the lists. That makes it easier to keept 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 *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` `\bothheadingname` The headings of the lists are stored in `\symheadingname` and `\subheadingname` and `\bothheadingname`. 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:

`\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:

`\dontmarkasused {` *macroname* `}`

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

Example:

```
\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:

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

2 Examples

These examples are supposed to illustrate the implications of `opensymdef` and `closesymdef`.

2.1 Example 1

Here, the definitions are in the preamble

```
\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}
```

Output:

$$E = m\ c^2$$

where E is the energy ...

List of Symbols (draft)

Symbol	Description
E	`\symE` – Energy (yes)
m	`\symm` – Mass (yes)
c	`\symc` – Speed of light (yes)

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]{symE}{E}
     \newsym[Mass]{symm}{m}
     \newsym[Speed of light]{symc}{c}

  \[\symE=\symm \symc^2\]

  where \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]{symE}{E}
     \newsym[Mass]{symm}{m}
     \newsym[Speed of light]{symc}{c}

  \[\symE=\symm \symc^2\]

  where \symE is the energy \ldots

  \closesymdef
\end{document}
```

3 Contact

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

e-mail: listofsymbols@gmx.de

4 The Code

```
\NeedsTeXFormat{LaTeX2e} \ProvidesPackage{listofsymbols}
\RequirePackage{ifthen} \RequirePackage{calc} \newboolean{b@nomencl}
\newboolean{b@final} \newboolean{b@Final} \newboolean{b@pageno}
\newboolean{b@xspace}
\DeclareOption{nomencl}{\setboolean{b@nomencl}{true}}
\DeclareOption{draft}{\setboolean{b@nomencl}{false}
\setboolean{b@final}{false}\setboolean{b@Final}{false}}
\DeclareOption{final}{\setboolean{b@nomencl}{false}
\setboolean{b@final}{true}\setboolean{b@Final}{false}}
\DeclareOption{Final}{\setboolean{b@nomencl}{false}
\setboolean{b@final}{true}\setboolean{b@Final}{true}}
\DeclareOption{pageno}{\setboolean{b@pageno}{true}}
\DeclareOption{nopageno}{\setboolean{b@pageno}{false}}
\DeclareOption{usexspace}{\setboolean{b@xspace}{true}}
\DeclareOption{noxspace}{\setboolean{b@xspace}{false}}

\ExecuteOptions{draft,nopageno,usexspace}
\ProcessOptions

\newlength{\symindent}
 %equal to second argument of \l@figure and \l@table:
\setlength{\symindent}{1.5em}
\newlength{\symwidth}
\setlength{\symwidth}{2.5cm}
\newlength{\sympagenowidth}

\ifthenelse{\boolean{b@nomencl}}
  {\RequirePackage{nomencl}}{}
\ifthenelse{\boolean{b@xspace}}
  {\RequirePackage{xspace}
  \newcommand{\spaceaftersym}{\xspace}}
  {\newcommand{\spaceaftersym}{}}
\ifthenelse{\boolean{b@pageno}}
  {\settowidth{\sympagenowidth}{9999}}
  {\setlength{\sympagenowidth}{0cm}}

%#1: sortkey
%#2: symbol
%#3: macroname
%#4: description
%#5: page number
\newcommand{\printsymline}[5]
{\expandafter\providecommand\expandafter{\csname#3include\endcsname}{no}
\ifthenelse{\boolean{b@final}
```

```
  \AND\(\expandafter\equal{\csname #3include\endcsname}{no}\OR\equal{#4}{}\)}
{\PackageInfo{listofsymbols}{symbol/subscript #3 has no or empty
  description or is not used: omitted}}
{\hspace*{\symindent}\makebox[2.5cm][l]{\ensuremath{#2}}
\parbox[t]{\textwidth-\symwidth-\sympagenowidth}
{\begin{raggedright}\strut%
\ifthenelse{\boolean{b@final}}{#4}
  {$\backslash$\texttt{#3} --- #4 (\csname #3include\endcsname)}%
\strut\end{raggedright}}%
\ifthenelse{\boolean{b@pageno}}{\hfill #5}{}%
\newline}}

\newcommand{\losstring}{}

%#1: sortkey
%#2: symbol
%#3: macroname
%#4: description
%#5: filehandle
\ifthenelse{\boolean{b@Final}}
{\newcommand{\addsymline}[5]{}
\newcommand{\opensymdef}{}
\newcommand{\closesymdef}{}}
{\newcommand{\opensymdef}
{\newwrite\@sym \immediate\openout\@sym=\jobname.sym
\newwrite\@sub \immediate\openout\@sub=\jobname.sub}
\newcommand{\closesymdef}
{\immediate\closeout\@sym
\immediate\closeout\@sub}
\newcommand{\addsymline}[5]
{\renewcommand{\losstring}{\string}
\immediate\write#5{\string\printsymline{\string#1}%
{\string#2}{\string#3}{#4}{\thepage}}
\renewcommand{\losstring}{}}}

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

%#1: description
%#2: macroname
```

```
%#3: symbol
\ifthenelse{\boolean{b@nomencl}}
{\newcommand{\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\expandafter{\csname#2\endcsname}
  {\relax\ensuremath{_{#3}}\spaceaftersym%
    \expandafter\protected@xdef\csname#2isused\endcsname {yes}}
\expandafter\newcommand\expandafter{\csname#2doc\endcsname}{#1}
\expandafter\newcommand\expandafter{\csname#2tabdoc\endcsname}
  {\ensuremath{#3} & #1}
\expandafter\newcommand\expandafter{\csname#2isused\endcsname}{no}}

%#1: description
%#2: macroname
%#3: symbol
\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}}}
{\newcommand{\symheading}
{\section*{\symheadingname\ (draft)}
\makebox[\symwidth+\symindent][l]{\bf Symbol}{\bf Description}
\ifthenelse{\boolean{b@pageno}}{\hfill{\bf Defined on page}}{}}
```

```
\newcommand{\subheading}
{\section*{\subheadingname\ (draft)}
\makebox[\symwidth+\symindent][l]{\bf Subscript}{\bf Description}
\ifthenelse{\boolean{b@pageno}}{\hfill{\bf Defined on page}}{}}}

\ifthenelse{\boolean{b@nomencl}}
{\makeglossary
\renewcommand{\nomname}{\symheadingname}
\setlength{\nomitemsep}{-1\parsep}
\newcommand{\listofsymbols}{\printglossary}
\newcommand{\listofsubscripts}{}}
{\newlength{\old@parskip}
\newlength{\old@parindent}
\newcommand{\listofsymbols} {
  \setlength{\old@parskip}{\parskip}
  \setlength{\parskip}{0pt}
  \setlength{\old@parindent}{\parindent}
  \setlength{\parindent}{0pt}
\symheading\par
\makeatletter
\IfFileExists{\jobname.syc}{\@input@{\jobname.syc}}{}
\IfFileExists{\jobname.sym}{\@input@{\jobname.sym}}{}
\makeatother
  \setlength{\parskip}{\old@parskip}
  \setlength{\parindent}{\old@parindent}}
\newcommand{\listofsubscripts} {
  \setlength{\old@parskip}{\parskip}
  \setlength{\parskip}{0pt}
  \setlength{\old@parindent}{\parindent}
  \setlength{\parindent}{0pt}
\subheading\par
\makeatletter
\IfFileExists{\jobname.suc}{\@input@{\jobname.suc}}{}
\IfFileExists{\jobname.sub}{\@input@{\jobname.sub}}{}
\makeatother
  \setlength{\parskip}{\old@parskip}
  \setlength{\parindent}{\old@parindent}}}

\ifthenelse{\boolean{b@nomencl}}
{\newcommand{\listofboth}{\listofsymbols}}
{\newcommand{\listofboth}
{\renewcommand{\symheading}{\subsection*{\symheadingname}}
\renewcommand{\subheading}{\subsection*{\subheadingname}}
\section*{\bothheadingname\ifthenelse{\boolean{b@final}}{}{ (draft)}}
\listofsymbols\listofsubscripts}}

\newcommand{\markasunused}[1]
  {\expandafter\protected@xdef\csname#1isused\endcsname {no}}

```

```
\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.sym}{\@input@{\jobname.sub}}{}
\immediate\closeout\@suc}

\endinput
```