The showkeys package*

David Carlisle

1995/04/25

1 Introduction

sec:intro

showkeys.sty modifies the \label, \ref, \pageref, \cite, and \bibitem commands so that the 'internal' key is printed. The package tries hard to position these labels so that the formatting of the rest of the document is unchanged. \label and \bibitem cause the key to appear in a box either in the margin, or in a TeX box of zero width, which may possibly over-print other text. The \ref, \pageref and \cite commands print their arguments in small type, raised just above the line, like this: sec:intro 1. This package works with the fleqn package, and the packages in the AMS-LaTeX collection.

2 Package Options

options

Some people have commented that the printing of the \ref and \cite keys is less useful than the printing of the \label keys and so showkeys now supports two options that can be given in the \usepackage command.:

notref to stop the redefinition of \ref and \pageref, and related commands from the varioref package.

notcite to stop the redefinition of \cite and related commands from the harvard and natbib packages.

So if the package is loaded with \usepackage[notref]{showkeys} then \ref will have its standard definition, but \label will print its key argument (usually in the margin).

3 More Examples

examples

The only other similar package that I could find in the macro index, DMJ:mi [3], was showlabels.sty, GN:sl [1]. After the first draft of this package was written, I found anon:sk [2] on my local installation! I think the current package is more robust than anon:sk [2], but I thought that showkeys was rather a good name, so I have stolen it for this file.

e^1

1. This has \label immediately after \item.

*This file has version number v3.03, last revised 1995/04/25.

e^2 2. This has the \label at the end.

A minipage :- Within environments like this minipage, We can not use \marginpar[1], so the appearance is slightly different. Here is that enumerate environment again:

m&e^1 1. This has \label immediately after \item.

m&e^2 2. This has the \label at the end.

Displayed math (without equation counter).

$$0 = 0$$ disp

Some text refering to the maths on page disp 2, and the item e^1 1.

If showkeys thinks that the current environment is going to produce an "equation number", then it does not show the label where the \label command occurs, but tries to put it in the margin, as shown with equation eq:xx 1. The package 'knows' about the standard equation and eqnarray environments, and also all the numbered alignment environments offered by the AMSLaTeX package, amsmath.

$$1 = 1 \tag{1}$$ eq:xx

$$\begin{aligned} 2 &= 2 && (2) \\ 3 &= 3 \\ 4 &= 4 && (3) \end{aligned}$$

eqnar:a

eqnar:b

Within a figure environment, the \label must not come before the \caption command. If you place \label inside the argument of \caption the label will be shown like this:

Figure 1: Within the caption argument. cap:a

If you place \label immediately after the \caption command it will be shown like this:

Figure 2: Immediately after the caption argument. cap:b

If you place the \label command at some random point after the \caption command, it may be shown like:

Figure 3: In vertical mode not immediately after a box.

cap:c

References

GN:sl [1] Gil Neiger, *showlabels.sty*, Undated package, similar to this one, but shows labels inline, affecting the formating of the document.

[1] Actually \marginpar is not used at all in this package now.

anon:sk [2] Annonymous, *showkeys.sty*, Package, dated 14 May 1988. Very similar to this one, also uses \marginpar in outer vertical mode.

DMJ:mi [3] David M. Jones, *TEX Macro Index*, A catalogue of TEX macros, including LATEX packages, available from all good TEX archives.

4 The Macros

```
⟨*package⟩
```

First we handle the options. Normally all related comands are defined to show there 'keys'. But since v3.03 one can specify:

`notref` to stop the redefinition of `\ref` (and `\pageref`, and related commands from varioref package)

`notcite` to stop the redefinition of `\cite` and related commands from the harvard and natbib packages.

```
\DeclareOption{notref}{\let\SK@ref\@empty}
\DeclareOption{notcite}{\let\SK@cite\@empty}
\ProcessOptions
```

\SK@label \SK@bibitem \SK@lbibitem

The saved original definitions

```
\let\SK@label\label
\let\SK@bibitem\@bibitem
\let\SK@lbibitem\@lbibitem
```

\label

The new definition, print the argument, and then do the old definition.

```
\def\label#1{\@bsphack\SK@\SK@@label{#1}\SK@label{#1}\@esphack}
```

\@bibitem \@lbibitem

For `\bibitem`, position the showkeys code as for a standard list with `\item` and `\label`.

```
\def\@bibitem#1{%
  \SK@bibitem{#1}\SK@\SK@@label{#1}\ignorespaces}
\def\@lbibitem[#1]#2{%
  \SK@lbibitem[#1]{#2}\SK@\SK@@label{#2}\ignorespaces}
```

\SK@

Grab hold of `#2` via `\meaning` so characters like `&` and `^` do not cause problems later, and pass the result on to the command `#1`.

```
\def\SK@#1#2{\def\@tempa{#2}\expandafter#1\meaning\@tempa\SK@}
```

\SK@@label

Strip off the initial segment of the `\meaning` output, and then put the rest either in a `\marginpar` or in a box of size 0pt, hopefully not disturbing the surrounding text.

```
\def\SK@@label#1>#2\SK@{%
```

Need to work globally as in some cases like alignments, and `fleqn`, the counter will be printed in a different group to the `\label` command.

```
  \gdef\SK@lab{\smash{\fbox{\reset@font\small\ttfamily#2}}}%
  \ifvmode
    \if@inlabel
```

If the \label is straight after \item (\bibitem is handled by this case as well) then the item label has not been added to the page yet. It is hanging around in the box \@labels waiting for the paragraph to start. So just need to attatch the label to this box.

```
      \global\setbox\@labels\hbox{%
        \llap{\SK@lab\SK@lab@relax
              \kern\@totalleftmargin\kern\marginparsep}%
        \box\@labels}%
```

The inner vertical mode cases are mainly designed to do the right thing with float captions, but seem to work OK in other cases as well.

```
    \else
      \ifinner
        \skip@=\lastskip\unskip
```

In inner vertical mode, attach the label to the right of the immediately preceding box, if it is a box before the current point. Otherwise just put it in a box of zero dimensions, with no interline skip. (This may slightly move the surrounding text.)

```
        \skip@\lastskip\unskip
        \setbox\z@\lastbox
        \ifvoid\z@
          \nointerlineskip\SK@labx
        \else
          \hbox{\box\z@\SK@labx}%
        \fi
        \vskip\skip@
      \else
```

In outer vertical mode, use a \vadjust at the start of the next paragraph (previously used \marginpar).

```
        \everypar\expandafter{\the\everypar
          \vadjust{\llap{\SK@lab\kern\marginparsep}}\SK@lab@relax}%
      \fi
    \fi
  \else
```

If we are in an numbered equation-style environment, do nothing as the code to print the number will also print the label, otherwise just stick the label at the current point, in a box of zero dimensions.

```
    \csname SK@\@currenvir\endcsname
    \ifSK@equation\else
      \ifmmode
        \SK@labx
      \else
```

Inner horizontal mode. Not much we can do, just stick it here.

```
        \ifinner
          \rlap\SK@lab
      \else
```

In outer horizontal mode use \vadjust to get to the margin.

```
          \vadjust{\llap{\SK@lab\kern\marginparsep}}%
        \fi
        \SK@lab@relax
      \fi
```

```
    \fi
  \fi}
```

`\iftagsleft@` Make sure that this AMSLaTeX command really is an `\if`..

```
\AtBeginDocument{%
  \let\SK@eqnnum\@eqnnum
  \def\@tempa{\let\iftagsleft@\iffalse}%
  \ifx\iftagsleft@\undefined\@tempa\fi%
```

`\tag@form@` `\eqnnum` Perhaps if `leqno` is operative, I should define `\@eqnnum` with the 'left' version, but it does not really matter.

```
  \let\SK@tagform@\tagform@
  \iftagsleft@
    \def\tagform@#1{%
      \ifx\df@label\@empty
        \SK@lab@relax
      \else
        \expandafter\SK@@label\meaning\df@label\SK@
      \fi
      \llap{\SK@lab\kern\marginparsep}%
      \SK@lab@relax\SK@tagform@{#1}}%
    \def\@eqnnum{%
      \llap{\SK@lab\kern\displaywidth\kern\marginparsep}%
      \SK@lab@relax\SK@eqnnum}%

  \else
    \def\tagform@#1{%
      \ifx\df@label\@empty
        \SK@lab@relax
      \else
        \expandafter\SK@@label\meaning\df@label\SK@
      \fi
      \SK@tagform@{#1}
      \rlap{\kern\marginparsep\SK@lab}\SK@lab@relax}%
    \def\@eqnnum{\SK@eqnnum\rlap{\kern\marginparsep\SK@lab}%
                      \SK@lab@relax}%
  \fi}
```

`\SK@labx` Print the label, and then globally reset the print command to `\relax`.

```
\def\SK@labx{\rlap\SK@lab\global\let\SK@lab\relax}
```

`\SK@lab@relax` Clear the label.

```
\def\SK@lab@relax{\global\let\SK@lab\relax}\SK@lab@relax
```

`\SK@equation` `\SK@eqnaray` `\SK@align` `\SK@alignat` `\SK@xalignat` `\SK@xxalignat` `\SK@gather` `\SK@multline` `\SK@flalign` The following environments print an equation number, so `\label` should not print its argument at the point where it appears. Note this will fail to show the label if you are in an `eqnarray` environment, and use `\label` together with `\nonumber` This might just about make sense if you are going to use `\pageref`, but that is too bad…

```
\newif\ifSK@equation
\let\SK@equation\SK@equationtrue
\let\SK@eqnarray\SK@equationtrue
\let\SK@align\SK@equationtrue
\let\SK@alignat\SK@equationtrue
```

```
\let\SK@xalignat\SK@equationtrue
\let\SK@xxalignat\SK@equationtrue
\let\SK@gather\SK@equationtrue
\let\SK@multline\SK@equationtrue
\let\SK@flalign\SK@equationtrue
```

The next section redefines `\ref` and `\pageref` (unless the `notref` otption was given).

```
\ifx\SK@ref\@empty\else
```

`\ref` `\pageref` Save the redefinition to `\begin{document}` so that this package can work with packages that redefine `\cite`. Tested with harvard and natbib packages. Also add code at this point to support varioref.

```
\AtBeginDocument{%
  \let\SK@ref\ref
  \let\SK@pageref\pageref
  \DeclareRobustCommand\ref[1]{\SK@\SK@@ref{#1}\SK@ref{#1}}%
  \DeclareRobustCommand\pageref[1]{\SK@\SK@@ref{#1}\SK@pageref{#1}}%
  \ifx\vpageref\@undefined\else
```

varioref support.

```
    \let\SK@@@vpageref\@@vpageref
    \def\@@vpageref#1[#2]#3{{%
      \let\label\SK@label\let\ref\SK@ref\let\pageref\SK@pageref
      \leavevmode\unskip\SK@\SK@@ref{#3}\SK@@@vpageref#1[#2]{#3}}}%
    \DeclareRobustCommand\vref[1]{%
      \unskip~\ref{#1}%
      {\let\label\SK@label\let\ref\SK@ref\let\pageref\SK@pageref
       \SK@@@vpageref\unskip[\unskip\space]{#1}}}%
  \fi}
```

```
\fi
```

Now redefine `\cite` unless `notcite` option given.

```
\ifx\SK@cite\@empty\else
```

`\cite`

```
\AtBeginDocument{%
  \ifx\HAR@checkdef\@undefined
```

Standard (non-harvard) support, including extra cite commands from natbib.

```
    \let\SK@cite\cite
    \def\cite#1#{\SK@citea{#1}}%
    \let\SK@citeauthor\citeauthor
    \def\citeauthor#1{\SK@\SK@@ref{#1}\SK@citeauthor{#1}}%
    \let\SK@citefullauthor\citefullauthor
    \def\citefullauthor#1{\SK@\SK@@ref{#1}\SK@citefullauthor{#1}}%
    \let\SK@citeyear\citeyear
    \def\citeyear#1{\SK@\SK@@ref{#1}\SK@citeyear{#1}}%
  \else
```

In the harvard style do *not* redefine individual cite commands. Just redefine one internal command that is used in all the citation forms.

```
    \let\SK@HAR@checkdef\HAR@checkdef
    \def\HAR@checkdef#1#2{%
```

```
      \expandafter\SK@\expandafter\SK@@ref\expandafter{#1}%
      \SK@HAR@checkdef{#1}{#2}}%
      \expandafter\let\expandafter
         \SK@HAR@bi\csname\string\harvarditem\endcsname
      \expandafter\def\csname\string\harvarditem\endcsname[#1]#2#3#4{%
        \SK@\SK@@label{#4}\SK@HAR@bi[#1]{#2}{#3}{#4}}%
  \fi}
```

```
\def\SK@citea#1#2{%
  \SK@\SK@@ref{#2}\SK@cite#1{#2}}
```

```
\fi
```

`\SK@@ref` This is much simpler than the printing of the label, as we know that we can be in horizontal mode.

```
\def\SK@@ref#1>#2\SK@{%
  \leavevmode\vbox to \z@{\vss
    \rlap{\vrule\raise .75em
       \hbox{\underbar{\reset@font\footnotesize\ttfamily#2}}}}}
```

```
⟨/package⟩
```