

Non-Floating Margin Notes with `marginnote` Package*

Markus Kohm[†]

2006/02/06

Abstract

In L^AT_EX the command `\marginpar[⟨left⟩]{⟨right⟩}` might be used to create a note in the margin. But there is a problem with this command: it creates a special kind of float. For this it cannot be used e.g. at floats or footnotes. Package `marginnote` supports another command `\marginnote` to create notes in the margin. This does not use a kind of float and for this does not have the disadvantage of `\marginpar`. But there might be other problems ...

1 How to Use `marginnote` Package

First of all you have to load. You may use:

```
\usepackage{marginnote}
```

to do so.

```
\marginnote
```

The command `\marginnote[⟨left⟩]{⟨right⟩}[⟨voffset⟩]` may be used to set a margin note using `marginnote`. The first optional argument and the mandatory argument are same using `\marginpar` from the L^AT_EX kernel. Even `\reversemarginpar` will be considered. The note `⟨left⟩` or `⟨right⟩` will be put at the current vertical position. Second optional argument `⟨voffset⟩` may be used to adjust the vertical position of the margin note. Use a negativ dimension to move it up or a positiv dimension to move it down.

```
\marginnoteleftadjust  
\marginnoterightadjust
```

At some environments, e.g. `framed` from the `framed` package the horizontal placement of the margin notes are not correct. In this case you may redefine `\marginnoteleftadjust` and `\marginnoterightadjust` to fix this. Note that these are macros not lengths! So you have to use `\renewcommand`, `\def` or `\let` to change them. You may e.g. use

```
\begingroup  
  \makeatletter  
  \g@addto@macro\framed{%  
    \let\marginnoteleftadjust\FrameSep  
    \let\marginnoterightadjust\FrameSep  
  }  
\endgroup
```

*This file has version number v1.0a, last revised 2006/02/06.

[†]Email: komascript@gmx.info

at your preamble after loading package `framed` to fix the problem using `framed` environment.

`\raggedleftmarginnote`
`\raggedrightmarginnote`

These macros define how the margin note will be aligned. The defaults are:

- align margin notes at the left margin right to the margin,
- align margin notes at the right margin left to the margin.

You may change this using `\renewcommand`, e.g. use

```
\renewcommand*{\raggedleftmarginnote}{}
\renewcommand*{\raggedrightmarginnote}{\centering}
```

to get justified text at the left and centered text at the right margin.

`\marginfont`

This macro defines the font that will be used to set margin notes. The default is `\normalcolor`. You may use `\renewcommand` to change this, e.g. use

```
\renewcommand*{\marginfont}{\color{red}\sffamily}
```

to get red colored margin notes in sans serif font family. You need to load e.g. package `color` to use `\color`.

2 Known Problems Using `marginnote`

At double side layout (e.g. using class option `twoside`) `\marginnote` needs to know the number of the current page to decide whether the page is odd or even and so whether to use left or right margin. \LaTeX uses an asynchronous output. Because of this counter `page` should not be used to get the number of the current page unless you are at page head or foot. To solve the problem `marginnote` uses a mechanism similar to labels. But this means, that the correct margin won't be known at this \LaTeX run but only at the next. So after adding or deleting a margin note or after each change of page break you need two \LaTeX runs to get all margins right.

The command `\marginnote` uses `\strut` and `\vadjust` to put the margin note at the correct position. But under some circumstances this may fail. You may adjust the vertical position of the margin note using the second optional argument of `\marginnote`. Sometimes even the text outside `\marginnote` will be moved because of using `\marginnote`. There's currently no other solution for this problem then moving the `\marginnote` command.

Note: The margin note will be placed at the current vertical line. This means, if you are using two `\marginnote` commands at the same line, they will be put on the same place. This is not a bug but a feature!

No page break may occur inside a margin note created with `\marginnote`.

`\marginnote` somewhat different from `\marginpar` if used immediately after `\item`. This is not a bug, it's a feature!

With math `\marginnote` may work or may not depending on the math environment.

3 Implementation

First process the options (no options are defined).

```
1 \ProcessOptions\relax
```

`\newmarginnote` We need a macro to define a new note at the aux file. This will be done using the mechanism of L^AT_EX that is used for `\newlabel`. But we use another prefix. This will result in the usual “Labels(s) may have changed. Rerun to get cross-references right.” if a margin note is new or have moved to another page.

```

2 \newcommand*{\newmarginnote}{\@newl@bel{mn}}

```

`\@mn@margintest` Macro `\@margintest` does the complete test, which margin to use. The result may be found at `\if@tempswa`. To avoid changes on the last page if there is a new note on the first page, try to count the notes by page. We know that this can not be successfull, but never the less it may be a good try. `\@mn@thispage` saves the page number of the last usage of `\@mn@margintest`. `\@mn@atthispage` saves the number of margin note at this page. But we need to know the absolut page number to do this. So we increase the absolut page number `mn@abspage` at every `\@outputpage`.

```

3 \newcommand*{\@mn@thispage}{%
4 \let\@mn@thispage\@empty
5 \newcounter{mn@abspage}
6 \AtBeginDocument{\setcounter{mn@abspage}{1}%
7 \g@addto@macro\@outputpage{\stepcounter{mn@abspage}}}%
8 \newcommand*{\@mn@margintest}{%

```

Number of the next margin note at this page.

```

9 \expandafter\ifx\csname @mn@thispage\endcsname\@empty
10 \gdef\@mn@atthispage{1}%
11 \else\expandafter\ifnum \@mn@thispage=\value{mn@abspage}%
12 \begin{group}
13 \@tempcnta\@mn@atthispage\advance\@tempcnta by \@ne
14 \xdef\@mn@atthispage{\the\@tempcnta}%
15 \end{group}
16 \else
17 \gdef\@mn@atthispage{1}%
18 \fi
19 \fi
20 \xdef\@mn@thispage{\themn@abspage}%

```

Use the number of the page and the number of the margin note at this page to save the real number of this page at the aux file.

```

21 \protected@write\@auxout{\let\themn@abspage\relax}{%
22 \string\newmarginnote{note.\@mn@thispage.\@mn@atthispage}{\themn@abspage}%
23 }%

```

If we are not in two side mode, we are on a odd page.

```

24 \if@twoside

```

If the margin note label was not defined, it seams to be new. In this case the absolut page number will be used for the test instead of the saved real page number.

```

25 \ifundefined{mn@note.\@mn@thispage.\@mn@atthispage}{%
26 \PackageInfo{marginnote}{Suggest that margin
27 note \@mn@thispage.\@mn@atthispage\space will be on\MessageBreak
28 absolute page \themn@abspage.\MessageBreak
29 This may be wrong}%
30 \ifodd\value{mn@abspage}\@tempswatrue\else\@tempswafalse\fi
31 }{%

```

```

32     \PackageInfo{marginnote}{Margin note \@mn@thispage.\@mn@atthispage\space
33     is on absolute page \@nameuse{mn@note.\@mn@thispage.\@mn@atthispage}}}%
34     \ifodd\@nameuse{mn@note.\@mn@thispage.\@mn@atthispage}
35         \@tempwattrue
36     \else
37         \@tempwafalse
38     \fi
39 }%
40 \else
41     \PackageInfo{marginnote}{right page because not two side mode}%
42     \@tempwattrue
43 \fi
44 }

```

```

\marginnote Command \marginnote is the main macro of the package. The others are helpers
\@mn@marginnote to manage the optional arguments.
\@mn@@marginnote 45 \newcommand*{\marginnote}{%
\@mn@@@marginnote 46 \@dblarg\@mn@marginnote
47 }
48 \newcommand{\@mn@marginnote}[2][ ]{%
49     \ifhmode
50         \@bsphack
51         \begingroup
52         \ifdim\@savsk>\z@ \else
53             \def\:{\@xifnch}\expandafter\def\:{ \futurelet\@let@token\@ifnch}%
54         \fi
55     \else
56         \begingroup
57     \fi
58     \@ifnextchar [{\@mn@@marginnote[#1][#2]}{\@mn@@marginnote[#1][#2][\z@]}%
59 }
60 \newcommand{\@mn@@marginnote}{}
61 \def\@mn@@marginnote[#1]#2[#3]{%
62     \endgroup

```

In horizontal mode the space hack of the L^AT_EX kernel will be used. In vertical mode this should not be used.

```

63     \ifhmode
64         \@mn@@marginnote[#1][#2][#3]%
65         \@esphack
66     \else
67         \@mn@@marginnote[#1][#2][#3]%
68     \fi
69 }
70 \newcommand{\@mn@@@marginnote}{}
71 \def\@mn@@@marginnote[#1]#2[#3]{%

```

All changes (but change of counters that are global because of using the L^AT_EX commands to change them an \gdef and \xdef) should be local. In h-mode a \strut will be used to fix base line. The margin note will be put to vertical list using \vadjust. This also means that we are one line to deep. This will be corrected later using negativ kern. In v-mode we use a special kind of vbox to simply set everything. Math mode should behave like v-mode. And if we are just after an item we have to leave v-mode first.

```

72 \begingroup
73 \ifmode\strut\let\@tempa\vadjust\else
74 \if@inlabel\leavevmode\fi
75 \ifhmode\strut\let\@tempa\vadjust\else\let\@tempa\mn@vlap\fi
76 \fi
77 \@tempa{%

```

Everything will be put upwards using a vbox with zero height and depth and \vss. At this box the margin test will be done. If csreversemargin was used, the logic switches. Then the note will be placed to the margin.

```

78 \vbox to\z@{%
79 \vss
80 \@mn@margintest
81 \if@reversemargin\if@tempswa
82 \@tempswafalse
83 \else
84 \@tempswatrue
85 \fi\fi
86 \if@tempswa
87 \rlap{\kern\textwidth\kern\marginparsep\kern\marginnoterightadjust
88 \vbox to\z@{\kern #3
89 \vbox to\z@{%
90 \hsize\marginparwidth

```

Here's the correction of the vertical position. The rest is simple.

```

91 \kern-\baselineskip\kern-\parskip
92 \marginfont\raggedrightmarginnote\hspace{\z@}\strut#2\endgraf
93 \vss}%
94 \vss}%
95 }%
96 \else
97 \llap{%
98 \vbox to\z@{\kern #3
99 \vbox to\z@{%
100 \hsize\marginparwidth

```

Same like above for left margins.

```

101 \kern-\baselineskip\kern-\parskip
102 \marginfont\raggedleftmarginnote\hspace{\z@}\strut#1\endgraf
103 \vss}%
104 \vss}%
105 \kern\marginparsep\kern\marginnoteleftadjust
106 }%
107 \fi
108 }%
109 }%
110 \endgroup
111 }

```

\marginnoterightadjust These may be used to define an automatic horizontal adjust. The default is zero.

```

\marginnoteleftadjust 112 \newcommand*{\marginnoterightadjust}{\}
113 \newcommand*{\marginnoteleftadjust}{\}
114 \let\marginnoterightadjust\z@
115 \let\marginnoteleftadjust\z@

```

`\mn@vlap` This macro is used to set a vertical box without size at vertical mode.

```

116 \newcommand{\mn@vlap}[1]{%
117   \setbox\@tempboxa\vbox to \ht\strutbox{#1\vss}%
118   \box\@tempboxa\vskip-\baselineskip\ifhmode\typeout{FALSCH!}\fi
119 }
```

`\marginfont` These are very simple. A class may also define `\marginfont`. Use this if available.

`\raggedleftmarginnote` I don't use `\let` for the definitions of the ragged macros, so the meaning may

`\raggedrightmarginnote` change loading e.g. package `ragged2e`.

```

120 \providecommand*\marginfont{}
121 \newcommand*\raggedleftmarginnote{\raggedleft}
122 \newcommand*\raggedrightmarginnote{\raggedright}
```

Change History

v1.0a	documentation fixed [thanks to	
General: Example to macros	Susumu Tanimura].	2
<code>\raggedleftmarginnote</code> and	<code>\marginfont</code> : Use <code>\providecommand</code>	
<code>\raggedrightmarginnote</code> at	to define it.	6