

A package for rotated objects in L^AT_EX^{*}

Robin Fairbairns[†] Sebastian Rahtz Leonor Barroca

printed June 6, 2008

Contents

1	Introduction	1
2	Usage	2
2.1	Package options	2
2.2	Float environments	2
2.3	Other environments and commands	2
2.4	Positioning	2
3	Setup	3
4	Turning and rotation environments	4
5	Sideways figures and tables	5
5.1	Rotated captions only	9
6	History	9

Abstract

This article documents a L^AT_EX package, ‘rotating.sty’, which performs most sorts of rotation one might like, including rotation of complete floating figures and tables.

1 Introduction

The package provides:

- two new environments, `sidewaystable` and `sidewaysfigure`, each of which produces a single page-size float with contents rotated ± 90 degrees; and
- a variety of other rotation-related commands and environments.

^{*}This file has version number v2.15a, last revised 2008/06/06

[†]Contact: rf10@cam.ac.uk

2 Usage

2.1 Package options

Sideways figures and tables always take up the whole page. In single-sided documents, they may be rotated so that the bottom of the figures is on the left (package option ‘counterclockwise’) or the right (package option ‘clockwise’). The default is to turn so that the bottom is on the right (option ‘clockwise’).

Option ‘anticlockwise’ is an alias for ‘counterclockwise’.

If the ‘twoside’ option has been given to the main document class (either explicitly, or implicitly as in the default for book class), the package will rotate sideways figures according to the page number (this requires at least two passes through L^AT_EX). If you want the ‘twoside’ option, but want the figures always in one direction, use the ‘figuresright’ or ‘figuresleft’ options to the package.

The package can produce a lot of logging information; the amount of information is controlled by the package options ‘quiet’ (fewest messages; default), ‘log’ and ‘chatter’ (most messages).

All other options are passed to the `graphicx` package when it is loaded to provide rotation functions.

2.2 Float environments

The environments `sidewaystable` and `sidewaysfigure` introduce landscape-form floating tables and figures, respectively. (Each of the environments has a “starred” version, such as `sidewaystable*`, for a single-column float in a double-column area of the document.)

New rotated environments may be declared using the combined facilities of the float and rotfloat packages.

2.3 Other environments and commands

The package provides other L^AT_EX environments:

`sideways` prints the contents of the environment turned through 90 degrees counterclockwise;

`turn` prints the contents turned through an arbitrary angle (the argument to the environment);

`rotate` prints the contents turned through an arbitrary angle but does *not* leave any space for the result

The command `\turnbox{⟨angle⟩}{⟨matter to turn⟩}` is a macro version of the `rotate` environment.

A set of examples is given in the file `examples.tex`

2.4 Positioning

Floats appear one to a page, and are positioned by spacer skips that appear (logically) above and below the floating object. The skips, `\rotFPtop` and `\rotFPbot`, are initialised from the standard L^AT_EX (internal) `\@fptop` and `\@fpbot` skips. As a result, by default, rotated floats appear horizontally centred on their float pages.

Some sensible values for the registers are:

<code>\rotFPtop</code>	<code>\rotFPbot</code>	Effect
0pt plus 1fil	0pt plus 1fil	figure/table appears in middle (default value)
0pt	0pt plus 1fil	figure/table appears with its top nearest the edge of the page
0pt plus 1fil	0pt plus 2fil	figure/table's bottom appears twice as far from the edge as the top does

3 Setup

Now we present the documented code. First, package options.

Note that the `clockwise` and `counterclockwise` options are present for compatibility only.

```

1 (*package)
2 \DeclareOption{clockwise}{%
3   \AtBeginDocument{\setkeys{Grot}{units=360}}%
4 }
5 \DeclareOption{counterclockwise}{%
6   \AtBeginDocument{\setkeys{Grot}{units=-360}}%
7 }
8 \DeclareOption{anticlockwise}{\ds@counterclockwise}

```

Control figure orientation

```

9 \DeclareOption{figuresleft}{%
10   \@rot@twosidefalse
11   \def\rot@LR{0}%
12 }
13 \DeclareOption{figuresright}{%
14   \@rot@twosidefalse
15   \def\rot@LR{-1}%
16 }

```

control messages:

```

17 \newif\ifrot@messages
18 \DeclareOption{quiet}{%
19   \rot@messagesfalse
20   \let\rot@message\@gobble % pro tem -- should suppress altogether
21 }
22 \DeclareOption{log}{%
23   \rot@messagestrue
24   \def\rot@message{\PackageInfo{rotating}}%
25 }
26 \DeclareOption{chatter}{%
27   \rot@messagestrue
28   \def\rot@message{\PackageWarning{rotating}}%
29 }

```

`\ifrot@twoside` A couple of commands for passing rotation direction around

```

\rot@LR
30 \newif\ifrot@twoside
31 \if@twoside
32   \@rot@twosidetrue
33 \else
34   \@rot@twosidefalse

```

```

35 \fi
36 \def\rot@LR{-1}

```

Pass any unknown options to the `graphicx` package, and set up defaults and process the options.

```

37 \DeclareOption*{\PassOptionsToPackage{\CurrentOption}{graphicx}}
38 \ExecuteOptions{clockwise,quiet}
39 \ProcessOptions

```

Other initialisation

```

40 \RequirePackage{graphicx}
41 \RequirePackage{ifthen}

```

\rotdriver The command `\rotdriver` allows a user to specify an initialisation file, a sort of non-automatically-loaded driver (in the graphics, hyperref sense).

```

42 \def\rotdriver#1{\makeatletter\input{#1.def}\makeatother}

```

The `r@tfl@t` counter is used when generating ‘labels’ for determining what side of the page the float is on, in twoside mode.

```

43 \newcounter{r@tfl@t}
44 \setcounter{r@tfl@t}{0}

```

Positioning skips (see above).

```

45 \newskip\rotFPtop \rotFPtop=\@fptop
46 \newskip\rotFPbot \rotFPbot=\@fpbot

```

4 Turning and rotation environments

sideways Environment to turn the contents through 90 degrees.

```

47 \def\sideways{%
48   \Grot@setangle{90}%
49   \setbox\z@\hbox\bgroup\ignorespaces}
50 \def\endsideways{%
51   \unskip\egroup
52   \Grot@x\z@
53   \Grot@y\z@
54   \Grot@box
55 }

```

turn Rotate the contents of the environment, leaving the appropriate space

```

56 \def\turn#1{%
57   \Grot@setangle{#1}%
58   \setbox\z@\hbox\bgroup\ignorespaces}
59 \def\endturn{%
60   \unskip\egroup
61   \Grot@x\z@
62   \Grot@y\z@
63   \Grot@box
64 }

```

rotate Rotate the contents of the environment, leaving *no space*.

```

65 \def\rotate#1{%
66   \Grot@setangle{#1}%

```

```

67 \setbox\z@\hbox\bgroup\ignorespaces}
68 \def\endrotate{%
69 \unskip\egroup
70 \Grot@x\z@
71 \Grot@y\z@
72 \wd0\z@\dp0\z@\ht0\z@
73 \Grot@box
74 }

\turnbox A macro version of the ‘rotate’ environment.
75 \def\turnbox#1#2{%
76 \Grot@setangle{#1}%
77 \setbox\z@\hbox{#2}%
78 \Grot@x\z@\Grot@y\z@
79 \wd0\z@\dp0\z@\ht0\z@
80 \Grot@box
81 }

```

5 Sideways figures and tables

Now for the macros to provide a complete environment for sideways figures and tables. We define two environments `sidewaysfigure` and `sidewaystable` that fit in with the normal table and figure floats. These are ‘fixed’ environments that just do 90 degree rotation, but it would be easy to parameterize this to do other rotations if needed (the mind boggles...)

```

\@rotfloat First a generalised ‘rotfloat’ environment. We need to intercept LATEX’s float
\@xrotfloat macros, in order to change the assumed width of a float being \columnwidth. We
\rot@float@box want it to work on a width of \textheight so that when we rotate the float,
it comes out the right height. This is not actually very satisfactory, since what
we really want is for rotated floats to occupy the space they actually use. The
captions are a problem — since they can precede the figure or table, we cannot
set them in a box of the right width (ie the height of the forthcoming object),
because it has not happened yet. The result of these difficulties is that rotated
figures always end up as full page figures.
82 \newsavebox\rot@float@box
83 \def\@rotfloat#1{%
84 \ifnextchar[%
85 {\@xrotfloat{#1}}%
86 {\edef\@tempa{\noexpand\@xrotfloat{#1}[\csname fps#1\endcsname]}\@tempa}%
87 }
88 \def\@xrotfloat#1[#2]{%
89 \@float{#1}[#2]%
90 }
91 }
92 }

Set the float contents in a box of width \textheight instead of \columnwidth.
90 \begin{lrbox}\rot@float@box
91 \begin{minipage}\textheight
92 }

\end@rotfloat We call LATEX’s \end@float macro having previously rotated the box \@currbox.
The rotation is either clockwise or anti-clockwise, depending on whether the page
is odd or even; in oneside mode it is always odd.
93 \def\end@rotfloat{%

```

If we are going to know whether pages are odd or even, we need to use the a variant `\pageref` mechanism and our own specialised labels.

```

94 \end{minipage}\end{lrbox}%
95 \stepcounter{r@tfl@t}%
96 \rot@label{RF\ther@tfl@t}%
97 \rot@pageref{RF\ther@tfl@t}{\R@@page}%
98 \rot@mess@toks{Adding sideways \c@type\space on page \R@@page\space}
99 \wd\rot@float@box\z@
100 \ht\rot@float@box\z@
101 \dp\rot@float@box\z@
102 \vbox to \textheight{%

```

We need to know for sure which direction rotation is going to be in, so locally reset the graphics units.

```

103 \setkeys{Grot}{units=360}%
104 \if@rot@twoside
105 \else
106 \let\R@@page\rot@LR
107 \fi
108 \ifthenelse{\isodd{\R@@page}}{%
109 \if@rot@twoside
110 \rot@mess@toks\expandafter{\the\rot@mess@toks right hand page}%
111 \fi
112 \vfill
113 \@@line{%
114 \hskip\rotFPtop
115 \rotatebox{90}{\box\rot@float@box}%
116 \hskip\rotFPbot
117 }%
118 }{%
119 \if@rot@twoside
120 \rot@mess@toks\expandafter{\the\rot@mess@toks left hand page}%
121 \fi
122 \@@line{%
123 \hskip\rotFPbot
124 \rotatebox{-90}{\box\rot@float@box}%
125 \hskip\rotFPtop
126 }%
127 \vfill
128 }%
129 \rot@message{\the\rot@mess@toks}
130 }%
131 \end@float
132 }

```

`\sidewaysfigure` The following definitions set up two environments, `sidewaystable` and `sidewaysfigure`, which uses this type of float. Naturally, users may need to change these to suit their local style. Both contribute to the normal lists of figures and tables.

```

\sidewaystable
\end{sidewaystable}
\sidewaysfigure
\end{sidewaysfigure}
133 \def\sidewaysfigure{\@rotfloat{figure}}
134 %
135 \let\endsidewaysfigure\end@rotfloat
136 %
137 \def\sidewaystable{\@rotfloat{table}}
138 \let\endsidewaystable\end@rotfloat

```

```

\@rotdblfloat Handling double column floats
\end@rotdblfloat 139 \def\@rotdblfloat{%
140 \if@twocolumn\let\reserved@a\@rotdbflt\else\let\reserved@a\@rotfloat\fi
141 \reserved@a}
142 \def\@rotdbflt#1{\@ifnextchar[{\@rotxdblfloat{#1}}{\@rotxdblfloat{#1}[tp]}}
143 \def\@rotxdblfloat#1[#2]{%
144 \hsize\textwidth\linewidth\textwidth
145 \@float{#1}[#2]%
146 \begin{lrbox}\rot@float@box
147 \begin{minipage}\textheight
148 }
149 \def\end@rotdblfloat{%

```

If we are going to know whether pages are odd or even, we need to use the `\pageref` mechanism, and labels. But Labels won't work *unless* the user has put in a caption. Beware!

```

150 \end{minipage}\end{lrbox}%
151 \stepcounter{r@tfl@t}%
152 \rot@label{RF\ther@tfl@t}%
153 \rot@pageref{RF\ther@tfl@t}{\R@@page}%
154 \rot@mess@toks{Adding sideways \c@type\space on page \R@@page\space}
155 \@tempdima\ht\rot@float@box
156 \advance\@tempdima by \dp\rot@float@box
157 % \ifrot@messages
158 % \rot@message{BOX wd: \the\wd\rot@float@box,
159 % ht: \the\ht\rot@float@box, dp: \the\dp\rot@float@box:
160 % so shift by .5 of \the\@tempdima}%
161 % \fi
162 \wd\rot@float@box\z@
163 \ht\rot@float@box\z@
164 \dp\rot@float@box\z@
165 \vbox to \textheight{%

```

We need to know for sure which direction rotation is going to be in, so locally reset the graphics units.

```

166 \setkeys{Grot}{units=360}%
167 \if@rot@twoside
168 \else
169 \let\R@@page\rot@LR
170 \fi
171 \ifthenelse{\isodd{\R@@page}}{%
172 \ifrot@messages
173 \if@rot@twoside
174 \rot@mess@toks\expandafter{\the\rot@mess@toks right hand page}%
175 \fi
176 \fi
177 \vfill
178 \@@line{%
179 \hskip\rotFPtop
180 \rotatebox{90}{\box\rot@float@box}%
181 \hskip\rotFPbot
182 }%
183 }{%
184 \ifrot@messages

```

```

185         \if@rot@twoside
186         \rot@mess@toks\expandafter{\the\rot@mess@toks left hand page}%
187 \fi
188         \fi%
189         \@@line{%
190             \hskip\rotFPbot
191             \rotatebox{-90}{\box\rot@float@box}%
192             \hskip\rotFPtop
193         }%
194         \vfill
195     }%
196     \rot@message{\the\rot@mess@toks}%
197 }%
198 \end@dblfloat
199 }

```

sidewaysfigure*

```

sidewaystable* 200 \newenvironment{sidewaystable*}
201                 {\@rot@dblfloat{table}}
202                 {\end@rot@dblfloat}
203 \newenvironment{sidewaysfigure*}
204                 {\@rot@dblfloat{figure}}
205                 {\end@rot@dblfloat}
206

```

`\rot@label` Note that we used `\rot@label`, not `\label`; this variant writes (just) the *true* page number, not the value of `\thepage`; this “true” value then needs special treatment
`\rot@thepage` in `\protected@write`, just as `\thepage` already has. `\rot@pageref{⟨generated label name⟩}`
`\rot@pageref` `{⟨cs to set to pageno⟩}` then returns the labelled page’s number (or 0
`\rot@protected@write` if label not yet defined). If label not defined, flags using `\rot@refundefinedtrue`
`\if@rot@refundefined` for end-document to pick up. (later...)

```

207 \def\rot@thepage{\@arabic\c@page}
208 \def\rot@label#1{\@bsphack
209   \rot@protected@write{\@auxout}{}%
210     {\string\newlabel{#1}{\rot@thepage}}%
211   \@esphack}
212 \def\rot@pageref#1#2{%
213   \expandafter\ifx\csname r@#1\endcsname\relax
214     \global\@rot@refundefinedtrue
215     \def#2{0}%
216   \else
217     \edef#2{\csname r@#1\endcsname}%
218   \fi
219 }
220 \long\def\rot@protected@write#1#2#3{%
221   \begingroup
222     \let\rot@thepage\relax
223     #2%
224     \let\protect\@unexpandable@protect
225     \edef\reserved@a{\write#1{#3}}%
226     \reserved@a
227   \endgroup
228   \if@nobreak\ifvmode\nobreak\fi\fi
229 }

```



```

230 \newif\if@rot@refundefined
231 \global\@rot@refundefinedfalse

\rot@mess@toks A token register to build up debugging messages
232 \newtoks\rot@mess@toks

```

5.1 Rotated captions only

`\rotcaption` Sometimes you may find that the rotation of complete figures does not give quite the right result, since they always take up the whole page. You may prefer to rotate the caption and the float contents separately within a conventional figure. Here we offer a suggestion for a `\rotcaption` command, which inserts the caption rotated by 90 degrees. It is essentially a copy of the normal captioning code. Packages which define the `\makecaption` command may also need to define `\makerotcaption`.

```

233 \def\rotcaption{\refstepcounter\@capytype\@dblarg{\@rotcaption\@capytype}}
234 \long\def\@rotcaption#1[#2]#3{%
235 \addcontentsline{\csname ext@#1\endcsname}{#1}{%
236 \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}}%
237 \par
238 \begingroup
239 \@parboxrestore
240 \normalsize
241 \makerotcaption{\csname fnum#1\endcsname}{#3}%
242 \endgroup}
243 \long\def\makerotcaption#1#2{%
244 \setbox\@tempboxa\hbox{#1: #2}%
245 \ifdim \wd\@tempboxa > .8\vsiz
246 \rotatebox{90}{%
247 \begin{minipage}{.8\textheight}#1: #2\end{minipage}%
248 }\par
249 \else%
250 \rotatebox{90}{\box\@tempboxa}%
251 \fi
252 \hspace{12pt}%
253 }
254 \end{package}

```

6 History

Version 2.0 is a complete re-write, with most of the work now being done by the $\text{\LaTeX 2}_{\epsilon}$ graphics package.

Version 2.1 provides a ‘clockwise’ option to reinstate the behaviour described in the ‘ \LaTeX Companion’

Version 2.2 just intercepts the standard float macros instead of copying and changing the. The ‘twoside’ option is obeyed.

Version 2.5 corrects problems in sideways figures.

Version 2.6 is a rewrite of the sideways floats via Frank Mittelbach (to whom many thanks for looking at the mangy code).

Version 2.7 is checked for \LaTeX of December 94, and adds the option of twoside behaviour independent of the general twoside.

Version 2.8 cleans up some mistakes pointed out by Harald Axel Sommerfeldt.
Version 2.9 cleans up some (more) mistakes pointed out by Harald Axel Sommerfeldt.

Version 2.13a permits positioning of rotated floats in the same way as they are positioned in ‘normal’ floats.

Version 2.14 is the first to be published anywhere as the outcome of maintenance by Robin Fairbairns.

Version 2.15 deals with page-numbering bug for auto-float-rotation, and tidying of messages; published to ctan

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols	I	
<code>\@makerotcaption</code> .. <u>233</u>	<code>\if@rot@refundefined</code> .. <u>207</u>	<code>\rotcaption</code> <u>233</u>
<code>\@rotdblfloat</code> <u>139</u>		<code>\rotdriver</code> <u>42</u>
<code>\@rotfloat</code> <u>82</u>	<code>\if@rot@twoside</code> ... <u>30</u>	S
<code>\@xrotfloat</code> <u>82</u>		<code>\sideways</code> <u>47</u>
E	R	<code>\sidewaysfigure</code> ... <u>133</u>
<code>\end@rotdblfloat</code> .. <u>139</u>	<code>\rot@float@box</code> <u>82</u>	<code>sidewaysfigure*</code> (environment) ... <u>200</u>
<code>\end@rotfloat</code> <u>93</u>	<code>\rot@label</code> <u>207</u>	<code>\sidewaystable</code> <u>133</u>
<code>\endsidewaysfigure</code> . <u>133</u>	<code>\rot@LR</code> <u>30</u>	<code>sidewaystable*</code> (environment) ... <u>200</u>
<code>\endsidewaystable</code> . <u>133</u>	<code>\rot@mess@toks</code> <u>232</u>	
environments:	<code>\rot@pageref</code> <u>207</u>	
<code>sidewaysfigure*</code> <u>200</u>	<code>\rot@protected@write</code> <u>207</u>	T
<code>sidewaystable*</code> . <u>200</u>	<code>\rot@thepage</code> <u>207</u>	<code>\turn</code> <u>56</u>
	<code>\rotate</code> <u>65</u>	<code>\turnbox</code> <u>75</u>