A package for rotated objects in \LaTeX*

Robin Fairbairns  Sebastian Rahtz  Leonor Barroca

printed October 31, 2022

Abstract

This article documents a \LaTeX{} 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, \texttt{sidewaystable} and \texttt{sidewaysfigure}, each of which produces a single page-size float with contents rotated $\pm90$ degrees; and

*This file has version number v2.16d, last revised 2016/08/11
• a variety of other rotation-related commands and environments.

Note that the package uses rotation facilities from the `graphicx` package. When generating DVI output, users should note that rotation is typically not visible in a DVI viewer: conversion to, and viewing, PostScript or PDF is necessary.

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 \LaTeX). 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 \LaTeX 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 `rotex.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 \LaTeX (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:

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

3 Setup

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

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

\begin{verbatim}
\langle∗ package⟩
\DeclareOption{clockwise}{% 
  \AtBeginDocument{\setkeys{Grot}{units=360}}% 
} \DeclareOption{counterclockwise}{% 
  \AtBeginDocument{\setkeys{Grot}{units=-360}}% 
} \DeclareOption{anticlockwise}{\ds@counterclockwise}
\DeclareOption{figuresleft}{% 
  \@rot@twosidefalse 
  \def\rot@LR{0} 
} \DeclareOption{figuresright}{% 
  \@rot@twosidefalse 
  \def\rot@LR{-1} 
}
\newif\ifrot@messages
\DeclareOption{quiet}{% 
  \rot@messagesfalse 
  \let\rot@message\@gobble % pro tem -- should suppress altogether 
} \DeclareOption{log}{% 
  \rot@messagestrue 
  \def\rot@message{\PackageInfo{rotating}}% 
} \DeclareOption{chatter}{% 
  \rot@messagestrue 
  \def\rot@message{\PackageWarning{rotating}}% 
}
\end{verbatim}

Control figure orientation

\begin{verbatim}
\DeclareOption{figuresleft}{% 
  \@rot@twosidefalse 
  \def\rot@LR{0} 
} \DeclareOption{figuresright}{% 
  \@rot@twosidefalse 
  \def\rot@LR{-1} 
}
\end{verbatim}

control messages:

\begin{verbatim}
\DeclareOption{quiet}{% 
  \let\rot@message\@gobble % pro tem -- should suppress altogether 
} \DeclareOption{log}{% 
  \rot@messagestrue 
  \def\rot@message{\PackageInfo{rotating}}% 
} \DeclareOption{chatter}{% 
  \rot@messagestrue 
  \def\rot@message{\PackageWarning{rotating}}% 
}
\end{verbatim}

3
A couple of commands for passing rotation direction around
\newif\ifrot\twoside
\if\twoside
\rot\twosidetrue
\else
\rot\twosidefalse
\fi
\def\rot\LR{-1}
Pass any unknown options to the graphicx package, and set up defaults and process the options.
\DeclareOption*{\PassOptionsToPackage{\CurrentOption}{graphicx}}
\ExecuteOptions{clockwise,quiet}
\ProcessOptions
Other initialisation
\RequirePackage{graphicx}
\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).
\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.
\newcounter{r@tfl@t}
\setcounter{r@tfl@t}{0}
Positioning skips (see above).
\newskip\rotFPtop \rotFPtop=\@fptop
\newskip\rotFPbot \rotFPbot=\@fpbot

4 Turning and rotation environments

sideways Environment to turn the contents through 90 degrees.
\def\sideways{\%}
\Grot@setangle{90}\
\setbox\z@\color@hbox\ignorespaces
\def\endsideways{\%
\unskip\color@endbox
\Grot@x\z@
\Grot@y\z@
\Grot@box
}

\def\turn{\%}
\Grot@setangle{#1}\
\setbox\z@\color@hbox\ignorespaces
\def\endturn{\%
\unskip\color@endbox
\Grot@x\z@

4
5 Sideways figures and tables

Now for the macros to provide a complete environment for sideways figures and tables. We define two environments \texttt{sidewaysfigure} and \texttt{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...)

\texttt{\@rotfloat\@xrotfloat\rot@float@box}

First a generalised ‘rotfloat’ environment. We need to intercept \LaTeX’s float macros, in order to change the assumed width of a float being \texttt{\columnwidth}. We want it to work on a width of \texttt{\textwidth} 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 \texttt{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.
Set the float contents in a box of width \textwidth instead of \columnwidth.
\begin{lrbox}\rot@float@box
\begin{minipage}\textwidth
\end@rotfloat

We call \LaTeX's \texttt{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.
\def\end@rotfloat{\If@twoside\let\R@@page\rot@LR\else\fi\ifthenelse{\isodd{\R@@page}}{\if@twoside\rot@mess@toks\expandafter{\the\rot@mess@toks (right hand page)}\fi\vfill\@@line{\hskip\rotFPtop\rotatebox{90}{\box\rot@float@box}\hskip\rotFPbot}\vfill}{\if@twoside\rot@mess@toks\expandafter{\the\rot@mess@toks (left hand page)}\fi\@@line{\hskip\rotFPbot\rotatebox{-90}{\box\rot@float@box}\hskip\rotFPtop}\vfill}\rot@message{\the\rot@mess@toks} \end@float
\end{minipage}\end{lrbox}

We need to know for sure which direction rotation is going to be in, so locally reset the graphics units.
\setkeys{Grot}{units=360}\if@rot@twoside\else\let\R@@page\rot@LR\fi\ifthenelse{\isodd{\R@@page}}{\if@rot@twoside\rot@mess@toks\expandafter{\the\rot@mess@toks (right hand page)}\fi}{\if@rot@twoside\rot@mess@toks\expandafter{\the\rot@mess@toks (left hand page)}\fi}\vfill\@@line{\hskip\rotFPtop\rotatebox{-90}{\box\rot@float@box}\hskip\rotFPbot}\vfill\rot@message{\the\rot@mess@toks} \end@float

\def\end@rotfloat{\If@twoside\let\R@@page\rot@LR\else\fi\ifthenelse{\isodd{\R@@page}}{\if@rot@twoside\rot@mess@toks\expandafter{\the\rot@mess@toks (right hand page)}\fi\vfill\@@line{\hskip\rotFPtop\rotatebox{90}{\box\rot@float@box}\hskip\rotFPbot}\vfill}{\if@rot@twoside\rot@mess@toks\expandafter{\the\rot@mess@toks (left hand page)}\fi\@@line{\hskip\rotFPbot\rotatebox{-90}{\box\rot@float@box}\hskip\rotFPtop}\vfill}\rot@message{\the\rot@mess@toks} \end@float
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.

```
134 \def\sidewaysfigure{\@rotfloat{figure}}
135 \let\endsidewaysfigure\end@rotfloat
138 \def\sidewaystable{\@rotfloat{table}}
139 \let\endsidewaystable\end@rotfloat
```

Handling double column floats
```
140 \def\rotdblfloat{\if@twocolumn\expandafter\@rotdbflt\else\expandafter\@rotfloat\fi}
143 \def\@rotdbflt#1{\ifnextchar[\@rotxdblfloat{#1}}{\@rotxdblfloat{#1}[tp]{}}
145 \@rotxdblfloat#1[#2]{\@float{#1}[#2]}
148 \begin{lrbox}\rot@float@box
150 \begin{minipage}\textheight}
```

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!
```
151 \end{minipage}\end{lrbox}
152 \stepcounter{r@tfl@t}
153 \rot@label{RF\ther@tfl@t}
154 \rot@pageref{RF\ther@tfl@t}{\R@@page}
155 \edef\@tempa{Adding sideways \@captype space on page \R@@page space}
156 \rot@mess@toks\expandafter{\@tempa}
157 \@tempdima\ht\rot@float@box
158 \advance\@tempdima by \dp\rot@float@box
159 \setkeys{Grot}{units=360} \if@rot@twoside \else \let\R@@page\rot@LR \fi
160 \ifthenelse{\isodd{\R@@page}}{\ifrot@messages}{\fi}
```

We need to know for sure which direction rotation is going to be in, so locally reset the graphics units.
```
168 \setkeys{Grot}{units=360} \if\rot@twoside \else \let\R@@page\rot@LR \fi
171 \ifthenelse{\isodd{\R@@page}}{\fi}{\f@rot@messages}
```

7
sidewaysfigure* (env.)

\newenvironment{sidewaystable*}{\@rotdblfloat{table}}{\end@rotdblfloat}
\newenvironment{sidewaysfigure*}{\@rotdblfloat{figure}}{\end@rotdblfloat}

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 for end-document to pick up. (later...)
\rot@protected@write#1#2#3{%
\begin{group}
\let\rot@thepage\relax
\let\protect\@unexpandable@protect
\edef\reserved@a{\write#1{#3}}\reserved@a
\end{group}
\if@nobreak\ifvmode\nobreak\fi\fi
\rot@refundefinedfalse
\rot@mess@toks A token register to build up debugging messages
\newtoks\rot@mess@toks

5.1 Rotated captions only

\rotcaption
\@makerotcaption

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 fig-
ure. 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.
\def\rotcaption{\refstepcounter{\@captype}\@dblarg{\@rotcaption{\@captype}}
\long\def\@rotcaption{[#2][#3]{%\addcontentsline{\csname ext@#1\endcsname}{#1}{%\protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}}%
\par
\begingroup
\@parboxrestore
\normalsize
\@makerotcaption{\csname fnum@#1\endcsname}{#3}\
\endgroup}
\long\def\@makerotcaption{\ifdim \wd\@tempboxa > .8\vsize
\rotatebox{90}{\begin{minipage}{.8\textheight}#1: #2\end{minipage}\
}\else\rotatebox{90}{\box\@tempboxa}\
\fi
\hspace{12pt}\
⟨/package⟩
6 Last-minute infrastructure

\color@hbox These macros aren’t provided in \LaTeX{}, by default (I seem to have assumed that\color@endbox
they were...)

\AtBeginDocument{\
providecommand\color@hbox{\hbox{\bgroup}}\n
providecommand\color@vbox{\vbox{\bgroup}}\n
providecommand\color@endbox{\egroup}\n}

7 History

Version 2.0 is a complete re-write, with most of the work now being done by the \LaTeX{}\texttt{2}\texttt{e} 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 them. 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

Version 2.16 uses colour boxes as necessary; published to ctan. Version 2.16a provides the colour box commands \AtBeginDocument.