The \texttt{ragged2e}-package\footnote{The version number of this file is v3.1, revision #(None), last revised (None).}

Martin Schröder

\url{https://gitlab.com/TExhackse/ragged2e}\footnote{maintained by Marei Peischl}

2021/12/15

Abstract

This package provides new commands and environments for setting ragged text which are easy to configure to allow hyphenation. An earlier attempt to do this was the style \texttt{raggedright} \cite{raggedright} by the same author.

Contents

1 The problem  
2 Old “solutions”  
\hspace{0.5cm} 2.1 \LaTeX  
\hspace{1cm} 2.2 plain  
3 Our solution  
\hspace{0.5cm} 3.1 The macros  
\hspace{1cm} 3.2 The parameters  
\hspace{1cm} 3.3 The environments  
4 Options  
5 Required packages  
6 The implementation  
\hspace{0.5cm} 6.1 Initial Code  
\hspace{1cm} 6.2 Declaration of options  
\hspace{1.5cm} 6.2.1 originalcommands option  
\hspace{1cm} 6.2.2 originalparameters option  
\hspace{1cm} 6.2.3 raggedrightboxes option  
\hspace{1cm} 6.2.4 footnotes option  
\hspace{1cm} 6.2.5 document option  
\hspace{1cm} 6.2.6 Other options  
\hspace{1cm} 6.3 Executing options  
\hspace{1cm} 6.4 Loading packages  
\hspace{1cm} 6.5 Allocations
1 THE PROBLEM

1.1 The problem

LaTeX has three commands (\centering, \raggedleft, and \raggedright) and three environments (center, flushleft, and flushright) to typeset ragged text. The environments are based upon the commands (center uses \centering, flushleft \raggedright, and flushright \raggedleft).

These commands have, however, one serious flaw: they render hyphenation almost impossible, and thus the text looks too ragged, as the following example shows:

\raggedright:

"The LaTeX document preparation system is a special version of Donald Knuth’s \TeX program. \TeX is a sophisticated program designed to produce high-quality typesetting, especially for mathematical text." [5, p. xiii]

\raggedleft:

"The \TeX document preparation system is a special version of Donald Knuth’s \TeX program. \TeX is a sophisticated program designed to produce high-quality typesetting, especially for mathematical text." [5, p. xiii]

2 Old “solutions”

2.1 \LaTeX

\LaTeX defines e. g. \raggedright as follows:

1 (\latex)

2 \DeclareRobustCommand
3 \raggedright\
4 \let\@centercr\
5 \flushglue
6 \rightskip\flushglue
7 \finalhyphendemerits=\z@\
8 \leftskip\z@skip
9 \parindent\z@}

Initially, \flushglue is defined as 10 \flushglue = Opt plus 1fil
11 (/latex)

Thus the \rightskip is set to Opt plus 1fil. Knuth, however warns [4, p. 101]:

“For example, a person can set \rightskip=Opt plus 1fil, and every line will be filled with space to the right. But this isn’t a particularly good way to make ragged-right margins, because the infinite stretchability will assign zero badness to lines that are very short. To do a decent job of ragged-right setting, the trick is to set \rightskip so that it will stretch enough to make line breaks possible, yet not too much, because short lines should be considered bad. Furthermore the spaces between words should be fixed so that they do not stretch or shrink.”

2.2 \texttt{plain}

\texttt{plain} \TeX defines an special version of \texttt{\raggedright}, which
3 OUR SOLUTION

Our solution operates the way Knuth describes it; but which can not be used with \LaTeX, because \LaTeX redefines \raggedright.

\begin{verbatim}
12 \def\raggedright{%
13 \rightskip\z@ plus2em
14 \spaceskip.3333em
15 \xspaceskip.5em\relax}
\end{verbatim}

\textit{plain} provides also a version of \raggedright for typewriter fonts

\begin{verbatim}
17 \def\ttraggedright{%
18 \tt
19 \rightskip\z@ plus2em\relax}
20 \texttt{/\plain}
\end{verbatim}

3 Our solution

Since the \textit{plain} solution can not be used with \LaTeX, we have to redefine it and make it possible to configure it for personal preferences.

3.1 The macros

\texttt{\Centering}, \texttt{\RaggedLeft}, and \texttt{\RaggedRight} can be used in the same way as \texttt{\centering}, \texttt{\raggedleft}, and \texttt{\raggedright}: Just type the command, and after that the whole text will be set centered, ragged-left or ragged-right.

For example, we switched on \texttt{\RaggedRight} on the top of this text, and consequently this text was set ragged-right.\textsuperscript{*}

\texttt{\justifying} switches back to justified text after ragged text has been switched on.

The new commands \texttt{\Centering}, \texttt{\RaggedLeft}, and \texttt{\RaggedRight} are fully compatible with their counterparts in \LaTeX, but implement the \textit{plain} solution and can be easily configured using the following parameters:

3.2 The parameters

\begin{tabular}{ll}
\hline
Command & Uses \\
\hline
\texttt{\Centering} & \texttt{\CenteringLeftskip}, \texttt{\CenteringRightskip}, \texttt{\CenteringParfillskip}, \texttt{\CenteringParindent} \\
\texttt{\RaggedLeft} & \texttt{\RaggedLeftLeftskip}, \texttt{\RaggedLeftRightskip}, \texttt{\RaggedLeftParfillskip}, \texttt{\RaggedLeftParindent} \\
\texttt{\RaggedRight} & \texttt{\RaggedRightLeftskip}, \texttt{\RaggedRightRightskip}, \texttt{\RaggedRightParfillskip}, \texttt{\RaggedRightParindent} \\
\texttt{\justifying} & \texttt{\justifyingParfillskip}, \texttt{\justifyingParindent} \\
\hline
\end{tabular}

All Parameters can be set with \texttt{\setlength}, e.g.

\begin{verbatim}
\setlength{\RaggedRightRightskip}{0pt plus 1em}
\end{verbatim}

\textsuperscript{*}For this documentation we also set \texttt{\RaggedRightRightskip} higher than usual (\texttt{0pt plus 2em} instead of \texttt{0pt plus 2em}) because of all the long command names which make linebreaking difficult.
sets \RaggedRightRightskip to $0pt \text{ plus } 1em$.

These are the \leftskip inserted by \Centering, \RaggedLeft, and \RaggedRight.

\textit{“leftskip (glue at left of justified lines)” [4, p. 274]}

\leftskip must be set to a finite value, to make hyphenation possible. Setting it to infinite values like $0pt \text{ plus } 1fil$ makes hyphenation almost impossible.

These are the \rightskip inserted by \Centering, \RaggedLeft, and \RaggedRight.

\textit{“rightskip (glue at right of justified lines)” [4, p. 274]}

\rightskip must be set to a finite value, to make hyphenation possible. Setting it to infinite values like $0pt \text{ plus } 1fil$ makes hyphenation almost impossible.

These are the \parfillskip inserted by \Centering, \RaggedLeft, \RaggedRight, and \justifying.

\textit{“parfillskip (additional \rightskip at end of paragraphs)” [4, p. 274]}

The normal setting for \parfillskip is $0pt \text{ plus } 1fil$; the parameters are provided for testing combinations of \{left\right\}skip and \parfillskip.

These are the \parindent used by \Centering, \RaggedLeft, \RaggedRight, and \justifying.

\textit{“parindent (width of indent)” [4, p. 274]}

\parindent is the indent of the first line of a paragraph and should be set to $0pt$, since indented lines in ragged text do not look good.

The parameters have the following initial setting:
3.3 The environments

Center Center is fully compatible with center, but uses \centering instead of centering.
FlushLeft FlushLeft is fully compatible with flushleft, but uses \raggedright instead of \raggedright.
FlushRight FlushRight is fully compatible with flushright, but uses \raggedleft instead of \raggedleft.
justify justify is like the other environments but uses \justifying.
E. g. FlushLeft can be used in the same way as flushleft:

\begin{FlushLeft}
⟨text, which is set ragged-right⟩
\end{FlushLeft}

4 Options

This package has the following options:

originalcommands The \LaTeX-commands \centering, \raggedleft, and \raggedright and the \LaTeX-environments center, flushleft, and flushright remain unchanged.
It is the default.

\footnote{†For proportional and monospaced fonts.}
newcommands  The \LaTeX-commands \texttt{\centering}, \texttt{\raggedleft}, and \texttt{\raggedright} and the \LaTeX-environments \texttt{center}, \texttt{flushleft}, and \texttt{flushright} are set equal to their counterparts defined by \texttt{ragged2e}. Thus \texttt{\raggedright} invokes \texttt{\RaggedRight}. The original commands can be accessed under the Names \texttt{\LaTeX\langle\text{original name}\rangle}, e. g. \texttt{\LaTeX\raggedright}.

originalparameters The parameters used by the commands implemented by \texttt{ragged2e} are initialized with the default settings used by \LaTeX.

newparameters The parameters used by the commands implemented by \texttt{ragged2e} are initialized with the default settings defined by \texttt{ragged2e}. It is the default.

raggedrightboxes All \texttt{\parbox}s, \texttt{\minipage}s, \texttt{\marginpar}s and \texttt{p}-columns of \texttt{tabular}s and \texttt{array}s are automatically set using \texttt{\RaggedRight}.

footnotes This options sets all footnotes ragged-right by loading the \texttt{footmisc}[2] package with the \texttt{ragged} option.

document This options sets the complete document ragged-right by executing a \texttt{\RaggedRight} at \texttt{\begin{document}} and the \texttt{raggedrightboxes} and the \texttt{footnotes} options.

All other options are passed to the \texttt{footmisc} package if the \texttt{footnotes} option is selected.

5  Required packages

This package requires the following packages:

everyset[8] (only if format older than 2021/01/05) It has been used to distinguish between monospaced and proportional fonts as long as the \LaTeX kernel did not provide the functionality with \texttt{lthooks}[6]. Formats newer than 2021/01/05 do no longer depend on \texttt{everyset}.

footmisc[2] It is used by the \texttt{footnotes} and the \texttt{document} options; at least version 5.00 (2002/08/28) is needed.

6  The implementation

6.1  Initial Code

\if@raggedtwoe@originalcommands is used to flag the use of the \texttt{originalcommands} or \texttt{newcommands} option.
6 THE IMPLEMENTATION

\ifraggedtwoeoriginalparameters
\ifraggedtwoeoriginalparameters is used to flag the use of the originalparameters or newparameters option.
23 \newif\ifraggedtwoeoriginalparameters

\ifraggedtwoefootmisc
\ifraggedtwoefootmisc is used to flag the use of the footnotes option.
24 \newif\ifraggedtwoefootmisc

6.2 Declaration of options

6.2.1 originalcommands option

The originalcommands and newcommands options control the meaning of the \LaTeX-commands for ragged text: If newcommands is used the \LaTeX-commands are set equal to the commands defined by \raggedtwoe.

25 \DeclareOption{OriginalCommands}{\@raggedtwoeoriginalcommandstrue}
26 \DeclareOption{originalcommands}{\@raggedtwoeoriginalcommandstrue}
27 \DeclareOption{NewCommands}{\@raggedtwoeoriginalcommandstruefalse}
28 \DeclareOption{newcommands}{\@raggedtwoeoriginalcommandstruefalse}

6.2.2 originalparameters option

The originalparameters and newparameters options control the defaults for the parameters used by the commands implemented by \raggedtwoe: If newparameters is used the parameters are set to the values defined by \raggedtwoe.

29 \DeclareOption{OriginalParameters}{\@raggedtwoeoriginalparameterstrue}
30 \DeclareOption{originalparameters}{\@raggedtwoeoriginalparameterstrue}
31 \DeclareOption{NewParameters}{\@raggedtwoeoriginalparameterstruefalse}
32 \DeclareOption{newparameters}{\@raggedtwoeoriginalparameterstruefalse}

6.2.3 raggedrightboxes option

The option raggedrightboxes sets all \parboxes, minipages, \marginpars and p-columns of tabulars and arrays using \RaggedRight. This is done by redefining \arrayparboxrestore.

\raggedtwo@raggedrightboxes@opt
\raggedtwo@raggedrightboxes@opt is the code executed via \DeclareOption.
33 \newcommand*{\@raggedtwo@raggedrightboxes@opt}{}

First we check if \arrayparboxrestore is unchanged.
34 \CheckCommand*{\@arrayparboxrestore}{}
35 \let\if\nobreak\iffalse
36 \let\if\noskipsec\iffalse
37 \let\par\@@par
38 \let\-\@dischyph
39 \let\'\@acci\let\"\@accii\let\\@acciii
Then we redefine it by removing the setting of `\leftskip`, `\rightskip`, \@rightskip and `\parfillskip` and instead calling `\RaggedRight`.

Now we self-destroy so the command can be called more than once without causing harm (and it also frees up some space).

Finally the declaration of the option.

6.2.4 `footnotes` option

The option `footnotes` just sets a flag (`\if@raggedtwoe@footmisc`) to load the `footmisc` package and passes the option `ragged` to it.

6.2.5 `document` option

The option `document` sets the complete document ragged-right by executing `\RaggedRight` via `\AtBeginDocument` and also executing the `raggedrightboxes` option.
\@raggedtwoe@abdhook is the code executed via \AtBeginDocument: Give a message on the terminal, execute \RaggedRight and self-destroy. We also make \@tocrmarg flexible; otherwise long lines in the table of contents (and similar tables) would not be broken because the spaceskip is rigid.

\newcommand{\@raggedtwoe@abdhook}{\PackageInfo{ragged2e}{ABD: executing \string\RaggedRight}}\RaggedRight\@ifundefined{@tocrmarg}{}{\edef\@tocrmarg{\@tocrmarg plus 2em}}\let\@raggedtwoe@abdhook\relax

Finally the declaration of the option.

\DeclareOption{document}{\@raggedtwoe@document@opt}

6.2.6 Other options

All unused options are passed to the footmisc package if the footnotes option is selected; otherwise the usual error is raised.

\DeclareOption*{\if\@raggedtwoe@footmisc\PassOptionsToPackage{\CurrentOption}{footmisc}\else\OptionNotUsed\fi}

6.3 Executing options

The default options are originalcommands and newparameters.

\ExecuteOptions{originalcommands,newparameters}\ProcessOptions\relax

6.4 Loading packages

We need the everyssel package for older kernels.
If the option footnotes is selected, we load the footmisc package after we are finished (footmisc detects our presence by looking for the definition of \RaggedRight, so we can not load it just now).

6.5 Allocations

First we allocate the parameters

\newlength{CenteringLeftskip}
\newlength{RaggedLeftLeftskip}
\newlength{RaggedRightLeftskip}
\newlength{CenteringRightskip}
\newlength{RaggedLeftRightskip}
\newlength{RaggedRightRightskip}
\newlength{CenteringParfillskip}
\newlength{RaggedLeftParfillskip}
\newlength{RaggedRightParfillskip}
\newlength{JustifyingParfillskip}
\newlength{CenteringParindent}
\newlength{RaggedLeftParindent}
\newlength{RaggedRightParindent}
\newlength{JustifyingParindent}

6.6 Initializations

Depending on \if@raggedtwoe@originalparameters we initialize the parameters with the values \LaTeX{} uses for its own commands or with our new parameters.

\if@raggedtwoe@originalparameters
\CenteringLeftskip\@flushglue
\RaggedLeftLeftskip\@flushglue
\RaggedRightLeftskip\z@skip
\CenteringRightskip\@flushglue
\RaggedLeftRightskip\z@skip
\RaggedRightRightskip\@flushglue
\CenteringParfillskip\z@skip
\RaggedLeftParfillskip\z@skip
\RaggedRightParfillskip\@flushglue
\CenteringParindent\z@
\RaggedLeftParindent\z@
\RaggedRightParindent\z@
\else
\CenteringLeftskip\z@\@plus\tw@ em
\RaggedLeftLeftskip\z@\@plus\tw@ em
\RaggedRightLeftskip\z@skip
\CenteringRightskip\z@\@plus\tw@ em
\RaggedLeftRightskip\z@skip
\RaggedRightRightskip\z@\@plus\tw@ em
\fi
6.7 Distinguishing between monospaced and proportional fonts

To set ragged text with proportional fonts and monospaced fonts correctly, we must distinguish between these two kinds of fonts \textit{everytime} a font is loaded. Otherwise the settings for e.g. a proportional fonts would be in effect if you start \texttt{RaggedRight} in \texttt{rmfamily} and then switch to \texttt{ttfamily}.

The goal is to have a rigid interword space in all fonts. TeX’s interword space is \texttt{\fontdimen2 plus \fontdimen3 minus \fontdimen4}. This can be overwritten by setting \texttt{\spaceskip} (space between words, if nonzero) and \texttt{\xspaceskip} (space at the end of sentences, if nonzero).

We do the setting with the help of \texttt{everyset[8]}, which allows us to define code which is (hopefully) executed after every fontchange in a LaTeX document.‡

\begin{verbatim}
\if@raggedtwoe@spaceskip
\if@raggedtwoe@spaceskip signals the use of commands defined by \ragged2e to the command inserted into \texttt{\selectfont}. It is set to true by these commands and restored to false by TeX when the scope of them ends.

\newif\if@raggedtwoe@spaceskip
\@raggedtwoe@everyselectfont\texttt{\@raggedtwoe@everyselectfont} is our code inserted into \texttt{\selectfont}.

\newcommand{\@raggedtwoe@everyselectfont}{%}
\if@raggedtwoe@spaceskip
If no command defined by \ragged2e is in use, we do nothing. But if it is, we look at \texttt{\fontdimen3} to see if the current font is monospaced or not.
\ifdim\fontdimen3\font=\z@\relax
If it is, we set \texttt{\spaceskip} to \texttt{0pt} so the interword space will be the one specified by the font designer – which is rigid anyway for monospaced fonts.
\else
\fi
\fi
\fi
\end{verbatim}

‡It is executed after every \texttt{\selectfont}, so if you stay within NFSS and don’t declare your fonts with commands like \texttt{\newfont} and then switch to them, it will work.
For proportional fonts we make the interword space rigid by setting \spaceskip to \fontdimen2. \fi

We have to reset the interword space if we are not active. \else \spaceskip\z@ \fi\}

If our kernel is new enough we use the kernel hook directly instead of the \everyisel macro. IfFormatAtLeastTF{2021/01/05}{\AddToHook{selectfont}{\@raggedtwoe@everyselectfont}}\EverySelectfont{\@raggedtwoe@everyselectfont}

\ref{6.8 The commands}

\@raggedtwoe@savedcr We save the definition of \ in \@raggedtwoe@savedcr. \let\@raggedtwoe@savedcr\}

\@raggedtwoe@saved@gnewline We save the definition of \@gnewline in \@raggedtwoe@saved@gnewline. \let\@raggedtwoe@saved@gnewline\@gnewline The following definition of a \@gnewline used by the ragged commands was suggested by Markus Kohm:

\newcommand*{\@raggedtwoe@gnewline}[1]{\ifvmode\@nolnerr\else\unskip\ifmmode\reserved@e{\reserved@f #1}\nobreak \hfil \break\else\reserved@e{\reserved@f #1}{\parskip\z@\par}\fi\fi\} \Centering \Centering first lets \ = \@centercr, but only if \ has its original meaning, otherwise \Center would not work inside environments like \texttt{tabular} etc., in which \ has a different meaning. It also sets \@gnewline to \@raggedtwoe@gnewline. Then, the \LaTeX\ and \TeX-parameters are set. \@rightskip is \LaTeX's version of \rightskip.
“Every environment, like the list environments, that set \rightskip to its ‘normal’ value set it to \@rightskip" [1]

Finally we signal the code inserted into \selectfont that we are active and call that code directly.

\begin{verbatim}
\DeclareRobustCommand{\Centering}{% 
  \ifx\ \@raggedtwoe@savedcr 
  \let\ \@centercr 
  \fi 
  \let\@gnewline\@raggedtwoe@gnewline 
  \leftskip\CenteringLeftskip 
  \@rightskip\CenteringRightskip 
  \rightskip\@rightskip 
  \parfillskip\CenteringParfillskip 
  \parindent\CenteringParindent 
  \@raggedtwoe@spaceskiptrue 
  \@raggedtwoe@everyselectfont 
  \}
\RaggedLeft
\RaggedLeft is like \Centering; it only uses other parameters. 
\DeclareRobustCommand{\RaggedLeft}{% 
  \ifx\ \@raggedtwoe@savedcr 
  \let\ \@centercr 
  \fi 
  \let\@gnewline\@raggedtwoe@gnewline 
  \leftskip\RaggedLeftLeftskip 
  \@rightskip\RaggedLeftRightskip 
  \rightskip\@rightskip 
  \parfillskip\RaggedLeftParfillskip 
  \parindent\RaggedLeftParindent 
  \@raggedtwoe@spaceskiptrue 
  \@raggedtwoe@everyselectfont 
  \}
\RaggedRight
\RaggedRight is like \Centering; it only uses other parameters. 
\DeclareRobustCommand{\RaggedRight}{% 
  \ifx\ \@raggedtwoe@savedcr 
  \let\ \@centercr 
  \fi 
  \let\@gnewline\@raggedtwoe@gnewline 
  \leftskip\RaggedRightLeftskip 
  \@rightskip\RaggedRightRightskip 
  \rightskip\@rightskip 
  \parfillskip\RaggedRightParfillskip 
  \parindent\RaggedRightParindent 
  \@raggedtwoe@spaceskiptrue 
  \@raggedtwoe@everyselectfont 
  \}
\justifying
\justifying switches back to the defaults used by \LaTeX\ for typesetting justified text.
\end{verbatim}
6.9 The environments

The environments Center, FlushLeft, and \FlushRight are implemented like their counterparts in \LaTeX: Start a trivlist and switch on the right command.

\begin{verbatim}
\newenvironment{Center}{% 
  \trivlist
  \Centering\item\relax
}\endtrivlist
\newenvironment{FlushLeft}{% 
  \trivlist
  \RaggedRight\item\relax
}\endtrivlist
\newenvironment{FlushRight}{% 
  \trivlist
  \RaggedLeft\item\relax
}\endtrivlist
\end{verbatim}

\begin{verbatim}
\newenvironment{justify}{% 
  \trivlist
  \justifying\item\relax
}\endtrivlist
\end{verbatim}

justify is similar to the other environments: Start a trivlist and use \justifying.

\begin{verbatim}
\newenvironment{justify}{% 
  \trivlist
  \justifying\item\relax
}\endtrivlist
\end{verbatim}

6.10 Overloading the \LaTeX-commands

If the option newcommands is used, we save the original \LaTeX-commands and environments for ragged text and overload them.

\begin{verbatim}
\if@raggedtwoe@originalcommands
\end{verbatim}
7 Acknowledgements

A first version of this package for \LaTeX2.09 was named raggedri [9]. Laurent Siebenmann (lcs@topo.math.u-psud.fr) with his style ragged.sty [10] provided the final impulse for this new implementation.

The code for justifying, justify and the overloading of \@arrayparboxrestore is incorporated from the raggedr [3] package by James Kilfinger (mapdn@csv.warwick.ac.uk).

Without the constant nagging of Rainer Sieger (rsieger@awi-bremerhaven.de) this package might not be.

Markus Kohm (markus.kohm@gmx.de) provided the code for \@gnewline.

Frank Mittelbach (frank.mittelbach@latex-project.org) provided the impetus for version 2.00.

Rolf Niepraschk (Rolf.Niepraschk@gmx.de) and Hubert Gäßlein found many bugs and provided fixes for them and code for new features.

Jordan Firth (jafirth@ncsu.edu) provided the final push for version 2.2.

References

(2) Robin Fairbairns. *footmisc* — a portmanteau package for customising footnotes in \LaTeX. \url{https://mirror.ctan.org/macros/latex/contrib/footmisc/}.

(3) James Kilfiger. \url{https://ctan.org/tex-archive/obsolete/macros/latex/contrib/raggedr.sty}. \LaTeX\ package.


(6) Frank Mittelbach. The \texttt{lthooks} package. \url{http://mirrors.ctan.org/macros/latex/base/lthooks-doc.pdf}

(7) Frank Mittelbach and Rainer Schöpf. The file \texttt{cmfonts.fdd} for use with \LaTeX. Part of the \LaTeX-distribution.

(8) Martin Schröder. The obsolete \texttt{everysel}-package. \url{http://mirrors.ctan.org/macros/latex/contrib/everysel/everysel.pdf}. \LaTeX\ package.

(9) Martin Schröder. The \texttt{raggedri} document option. Was in \url{http://mirrors.ctan.org/tex-archive/macros/latex209/contrib/raggedright}. \LaTeX2.09 style, outdated.

(10) Laurent Siebenmann. \texttt{ragged.sty}. \texttt{CTAN:tex-archive/macros/generic/ragged.sty}. generic macro file for \texttt{plain} and \texttt{\LaTeX}.

**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.

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Code Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>\verb</td>
<td>^</td>
</tr>
<tr>
<td>\verb</td>
<td>--</td>
</tr>
<tr>
<td>\verb</td>
<td>=</td>
</tr>
<tr>
<td>\verb</td>
<td>@par</td>
</tr>
<tr>
<td>\verb</td>
<td>@acci</td>
</tr>
<tr>
<td>\verb</td>
<td>@accii</td>
</tr>
<tr>
<td>\verb</td>
<td>@acciii</td>
</tr>
<tr>
<td>\verb</td>
<td>@arrayparboxrestore</td>
</tr>
<tr>
<td>\verb</td>
<td>@centercr</td>
</tr>
<tr>
<td>\verb</td>
<td>@dischyph</td>
</tr>
<tr>
<td>\verb</td>
<td>@flushglue</td>
</tr>
<tr>
<td>\verb</td>
<td>@nevaline</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
<tr>
<td>\verb</td>
<td>@ifundefined</td>
</tr>
</tbody>
</table>
Index

J
\justify (environment) .......... 5, 239
\justifying ............. 3, 210, 241
\JustifyingParfillskip ............. 4, 99, 140, 216
\JustifyingParindent 4, 99, 141, 217

L
\LaTeXcenter ............ 253
\LaTeXcentering ........... 247
\LaTeXflushleft ........... 255
\LaTeXflushright ......... 257
\LaTeXraggedleft .......... 248
\LaTeXraggedright ........ 249
\leftskip ............. 8, 44, 178, 189, 202, 213
\let .................. 4, 35, 36, 37, 38, 39, 51, 52, 53, 54, 55, 65, 76, 82, 157, 158, 173, 175, 186, 188, 199, 201, 211, 212, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264
\lineskip ................ 46, 61
\lineskiplimit ............. 47, 63
\linewidth ............ 42, 58

N
\newcommand .......... 33, 72, 78, 143, 159
\newenvironment .... 221, 227, 233, 239
\newif .................. 22, 23, 24, 142
\newlength ............ 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112
\nobreak ............ 165
\normalbaselineskip .... 48, 62
\normalineskip ......... 46, 61
\normalineskiplimit .... 47, 63

O
\OptionNotUsed ............ 89

P
\PackageInfo ............. 73
\par ............. 37, 53, 167
\parfillskip ....... 45, 179, 192, 205, 216
\parindent .............. 9, 40, 141, 180, 193, 206, 217
\parskip ................ 40, 56, 167
\PassOptionsToPackage .... 70, 87
\ProcessOptions .......... 93
\providecommand .......... 94

R
\RaggedLeft ............ 3, 184, 235, 251
\raggedleft ............ 248, 251
\RaggedLeftLeftskip ............. 4, 99, 115, 128, 189
\RaggedLeftParfillskip ............. 4, 99, 121, 134, 192
\RaggedLeftParindent ....... 4, 99, 124, 137, 193
\RaggedLeftRightskip ............. 4, 99, 118, 131, 190
\RaggedRight ............. 3, 60, 74, 167, 229, 252
\raggedright ............. 3, 13, 249, 252
\RaggedRightLeftskip ............. 4, 99, 116, 129, 202
\RaggedRightParfillskip ............. 4, 99, 122, 135, 205
\RaggedRightParindent ....... 4, 99, 125, 138, 206
\RaggedRightRightskip ............. 4, 99, 119, 132, 203
\relax ............. 16, 19, 65, 76, 82, 93, 145, 223, 229, 235, 241
\renewcommand ........... 50
\RequirePackage .......... 95, 97
\reserved@e ............ 165, 167
\reserved@f ............ 165, 167
\rightskip ............ 6, 14, 19, 44, 178, 191, 204, 215

S
\sloppy ............ 49, 64
\spaceskip ............ 15, 146, 148, 151
\string ............. 73

T
\thr@@ ............. 145
\trivlist ............. 222, 228, 234, 240
\tt ............. 18
\ttraggedright ............. 17
\tw@ ............. 127, 128, 130, 132, 148

U
\unskip ............. 163
\xspaceskip ............. 16

X
\z@skip ............. 7, 9, 14, 19, 40, 43, 59, 123, 124, 125, 127, 128, 130, 132, 136, 137, 138, 145, 146, 151, 167, 213, 214
\z@skip ............. 8, 40, 44, 56, 116, 118, 120, 121, 129, 131, 133, 134
\Z ............. 7, 9, 14,
Change History

v1.00  
General: New from \raggedright V 1.21  ................................. 1
v1.01  
General: Documentation improved  ................................. 1
v1.02  
General: Moved to LPPL  ............................................ 1
v2.00  
\@raggedtwoe@abedhook: New macro  ................................. 9
\@raggedtwoe@document@opt: New macro  ............................. 9
\@raggedtwoe@everyselectfont: Completely redesigned and removed  
  \RaggedSpaceskip and \RaggedXSpaceskip  ......................... 11
\@raggedtwoe@gnewline: New macro  ................................. 12
\@raggedtwoe@raggedrightboxes@opt: New macro  .................... 7
\@raggedtwoe@saved@gnewline: New macro  ............................ 12
\Centering: Call \@raggedtwoe@everyselectfont and switch  
  \gnewline ......................................................... 13
\JustifyingParfillskip: New macro  ................................. 10
\JustifyingParindent: New macro  ................................. 10
\RaggedLeft: Call \@raggedtwoe@everyselectfont and switch  
  \gnewline ......................................................... 13
\RaggedRight: Call \@raggedtwoe@everyselectfont and switch  
  \gnewline ......................................................... 13
\if@raggedtwoe@footmisc: New macro  ................................. 7
\justifying: New macro  ............................................. 13
General: Allow all-lowercase versions of options  .................. 7  
  Allow all-lowercase versions of options and removed documentation of 
  mixed-case versions. ............................................. 5
Incorporated \raggedr ................................................. 1
Load the \footmisc package ........................................... 9
New command \justifying ............................................. 3
New environment justify ............................................. 5
New option document ................................................. 6, 8
New option footnotes ................................................. 6, 8
New option raggedrightboxes ......................................... 6, 7
Pass all other options to \footmisc if it’s loaded ................. 9
Removed \RaggedSpaceskip and \RaggedXSpaceskip .................. 3, 10
Removed spaces and unneeded braces from \setlength; replaced plus 
  with \@plus ..................................................... 6
justify: New environment ............................................. 14
v2.01 \@raggedtwoe@everyselectfont: Removed the setting of \xspaceskip .......................... 11
v2.02 \@raggedtwoe@gnewline: Bugfix: \@nolerr \rightarrow \@nolnerr ................................. 12
General: Bugfix: \if@raggedtwoe@footmisc@true \rightarrow  
  \@raggedtwoe@footmisc@true ..................................... 8
Removed \setlength .................................................. 8
Use \@flushglue ..................................................... 6
FlushRight: Use \trivlist \ldots \endtrivlist instead of  
  \begin{trivlist} \ldots \end{trivlist} ............................. 14
justify: Use \trivlist \ldots \endtrivlist instead of \begin{trivlist} 
  \ldots \end{trivlist} ............................................. 14
v2.03
  General: Bugfix: footnotes was actually raggedrightboxes .................... 8
v2.04
  \raggedtwoe@abdhook: Set \@tocrmarg and use \PackageInfo .................... 9
  \raggedtwoe@everyselectfont: Reset \spaceskip when we are not active ...................... 12
  \raggedtwoe@raggedrightboxes@opt: The setting of \parindent is superfluous ..................... 8
  General: Initialize \JustifyingParindent with \parindent ..................... 10
  Insert missing \ ................................................................. 10
  Save more commands ......................................................... 14
v2.1
  \raggedtwoe@abdhook: bugfix: Use \@tocrmarg only if it's defined ........... 9
  \raggedtwoe@gnewline: Bugfix: handle math .................................... 12
  General: bugfix: \Flushleft instead of \FlushLeft (found by Berend Hasselman) .................................................. 14
  bugfix: Load footmisc directly and not via \AtEndOfPackage (bug found by Axel Sommerfeldt) ................................................................. 10
  document that document needs footmisc ...................................... 6
v2.2
  \raggedtwoe@raggedrightboxes@opt: Definition of \Arrayparboxrestore has changed ......................... 7
  General: Change maintenance status of package ............................ 2
  Move to git/gitlab, use svninfo2 ........................................... 1
  Require a new version of \LaTeX(2017/03/29) .................................. 1
v3.0
  General: Change maintenance status ............................................ 2
  document everyls is obsolete .................................................. 6
v3.00
  \raggedtwoe@everyselectfont: Use kernel hook if available .................. 12
  General: Remove the everyls package if kernel hooks are available .......... 9
v3.1
  \Centering: Robustify the user macros (Thanks to Markus Kohm for the hint) ................................................................. 13
  \RaggedLeft: Robustify the user macros ...................................... 13
  \RaggedRight: Robustify the user macros (Thanks to Markus Kohm for the hint) ................................................................. 13
  \justifying: Robustify the user macros (Thanks to Markus Kohm for the hint) ................................................................. 13
  General: Use the updated definition of \raggedright .......................... 2