The ragged2e-package*

Martin Schröder
https://gitlab.com/TeXhackse/ragged2e†

2023/06/22

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 raggedright [9] by the same author.

Contents

1 The problem **2**

2 Old "solutions" **2**
2.1 LaTeX . . . 2
2.2 plain . . . 2

3 Our solution **3**
3.1 The macros . . . 3
3.2 The parameters . . . 3
3.3 The environments . . . 5

4 Options **5**

5 Required packages **6**

6 The implementation **6**
6.1 Initial Code . . . 6
6.2 Declaration of options . . . 7
6.2.1 originalcommands option . . . 7
6.2.2 originalparameters option . . . 7
6.2.3 raggedrightboxes option . . . 7
6.2.4 footnotes option . . . 8
6.2.5 document option . . . 8
6.2.6 Other options . . . 9
6.3 Executing options . . . 9
6.4 Loading packages . . . 9
6.5 Allocations . . . 10

*The version number of this file is v3.6, revision #c0a7166, last revised 2023-06-22 10:55:19 +0200.

†maintained by Marei Peischl

6.6 Initializations 10
6.7 Distinguishing between monospaced and proportional fonts . 11
6.8 The environments 15
6.9 Overloading the LaTeX-commands 15

7 Acknowledgements 16

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]

`\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]

2 Old "solutions"

2.1 LaTeX

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

```
1 ⟨*latex⟩
2 \DeclareRobustCommand
3   \raggedright{%
4   \let\\\@centercr
5   \@rightskip\@flushglue
6   \rightskip\@rightskip
7   \finalhyphendemerits=\z@
8   \leftskip\z@skip
9   \parindent\z@}
```

Initially, `\@flushglue` is defined as

```
10 \@flushglue = 0pt plus 1fil
11 ⟨/latex⟩
```

Thus the `\rightskip` is set to `0pt plus 1fil`. Knuth, however warns [4, p. 101]:

> "For example, a person can set `\rightskip=0pt 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 infinte 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 plain

plain TeX defines an special version of `\raggedright`, which operates the way Knuth describes it; but which can not be used whith

LaTeX, because LaTeX redefines `\raggedright`.

```
12 ⟨*plain⟩
13 \def\raggedright{%
14     \rightskip\z@ plus2em
15     \spaceskip.3333em
16     \xspaceskip.5em\relax}
```

`plain` provides also a version of `\raggedright` for typewriter fonts

```
17 \def\ttraggedright{%
18     \tt
19     \rightskip\z@ plus2em\relax}
20 ⟨/plain⟩
```

3 Our solution

Since the `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

`\Centering` `\RaggedLeft` `\RaggedRight`

`\Centering`, `\RaggedLeft`, and `\RaggedRight` can be used in the same way as `\centering`, `\raggedleft`, and `\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 `\RaggedRight` on the top of this text, and consequently this text was set ragged-right.*

`\justifying`

`\justifying` switches back to justified text after ragged text has been switched on.

New feature 2003/01/04

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

3.2 The parameters

Command	Uses
`\Centering`	`\CenteringLeftskip`, `\CenteringRightskip`, `\CenteringParfillskip`, `\CenteringParindent`
`\RaggedLeft`	`\RaggedLeftLeftskip`, `\RaggedLeftRightskip`, `\RaggedLeftParfillskip`, `\RaggedLeftParindent`
`\RaggedRight`	`\RaggedRightLeftskip`, `\RaggedRightRightskip`, `\RaggedRightParfillskip`, `\RaggedRightParindent`
`\justifying`	`\JustifyingParfillskip`, `\JustifyingParindent`

All Parameters can be set with `\setlength`, e. g.

```
\setlength{\RaggedRightRightskip}{0pt plus 1em}
```

sets `\RaggedRightRightskip` to `0pt plus 1em`.

`\CenteringLeftskip` `\RaggedLeftLeftskip` `\RaggedRightLeftskip`

These are the `\leftskip`s inserted by `\Centering`, `\RaggedLeft`, and

*For this documentation we also set `\RaggedRightRightskip` higher than usual (`0pt plus 4em` instead of `0pt plus 2em`) because of all the long command names which make linebreaking difficult.

\RaggedRight.

> "\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 plus 1fil makes hyphenation almost impossible.

\CenteringRightskip
\RaggedLeftRightskip
\RaggedRightRightskip

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

> "\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 plus 1fil makes hyphenation almost impossible.

\CenteringParfillskip
\RaggedLeftParfillskip
\RaggedRightParfillskip
\JustifyingParfillskip

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

> "\parfillskip (additional \rightskip at end of paragraphs)" [4, p. 274]

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

\CenteringParindent
\RaggedLeftParindent
\RaggedRightParindent
\JustifyingParindent

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

> "\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:

Parameter	LaTeX setting	ragged2e setting[†]
\CenteringLeftskip	0pt plus 1fil	0pt plus 2em
\RaggedLeftLeftskip	0pt plus 1fil	0pt plus 2em
\RaggedRightLeftskip	0pt plus 0pt minus 0pt	0pt plus 0pt minus 0pt
\CenteringRightskip	0pt plus 1fil	0pt plus 2em
\RaggedLeftRightskip	0pt plus 0pt minus 0pt	0pt plus 0pt minus 0pt
\RaggedRightRightskip	0pt plus 1fil	0pt plus 2em
\CenteringParfillskip	0pt plus 0pt minus 0pt	0pt plus 0pt minus 0pt
\RaggedLeftParfillskip	0pt plus 0pt minus 0pt	0pt plus 0pt minus 0pt
\RaggedRightParfillskip	0pt plus 1fil	0pt plus 1fil
\CenteringParindent	0pt	0pt
\RaggedLeftParindent	0pt	0pt
\RaggedRightParindent	0pt	0pt
\JustifyingParfillskip		0pt plus 1fil
\JustifyingParindent		\parindent

3.3 The environments

`Center` (*env.*) Center is fully compatible with center, but uses `\Centering` instead of `\centering`.

`FlushLeft` (*env.*) FlushLeft is fully compatible with flushleft, but uses `\RaggedRight` instead of `\raggedright`.

`FlushRight` (*env.*) FlushRight is fully compatible with flushright, but uses `\RaggedLeft` instead of `\raggedleft`.

`justify` (*env.*) justify is like the other environments but uses `\justifying`.

New feature 2003/01/04

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.

[†]For proportional and monospaced fonts.

newcommands The LaTeX-commands `\centering`, `\raggedleft`, and `\raggedright` and the LaTeX-environments center, flushleft, and flushright are set equal to their counterparts defined by ragged2e. Thus `\raggedright` invokes `\RaggedRight`. The original commands can be accessed unter the Names `\LaTeX`⟨*original name*⟩, e. g. `\LaTeXraggedright`.

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

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

New feature 2003/01/18

raggedrightboxes All `\parbox`es, minipages, `\marginpar`s and p-columns of tabulars and arrays are automatically set using `\RaggedRight`.

New feature 2003/01/18

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

New feature 2003/01/18

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

All other options are passed to the footmisc package if the footnotes option is selected.

5 Required packages

This package requires the following packages:

everysel [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 lthooks [6]. Formats newer than 2021/01/05 do no longer depend on everysel.

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

6 The implementation

21 ⟨∗package⟩

6.1 Initial Code

`\if@raggedtwoe@originalcommands` `\if@raggedtwoe@originalcommands` is used to flag the use of the originalcommands or newcommands option.

22 `\newif\if@raggedtwoe@originalcommands`

\if@raggedtwoe@originalparameters `\if@raggedtwoe@originalparameters` is used to flag the use of the originalparameters or newparameters option.

```
\newif\if@raggedtwoe@originalparameters
```

\if@raggedtwoe@footmisc `\if@raggedtwoe@footmisc` is used to flag the use of the footnotes option.

```
\newif\if@raggedtwoe@footmisc
```

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 ragged2e.

```
\DeclareOption{OriginalCommands}{\@raggedtwoe@originalcommandstrue}
\DeclareOption{originalcommands}{\@raggedtwoe@originalcommandstrue}
\DeclareOption{NewCommands}{\@raggedtwoe@originalcommandsfalse}
\DeclareOption{newcommands}{\@raggedtwoe@originalcommandsfalse}
```

6.2.2 originalparameters option

The originalparameters and newparameters options control the defaults for the parameters used by the commands implemented by ragged2e: If newparameters is used the parameters are set to the values defined by ragged2e.

```
\DeclareOption{OriginalParameters}{\@raggedtwoe@originalparameterstrue}
\DeclareOption{originalparameters}{\@raggedtwoe@originalparameterstrue}
\DeclareOption{NewParameters}{\@raggedtwoe@originalparametersfalse}
\DeclareOption{newparameters}{\@raggedtwoe@originalparametersfalse}
```

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`.

\@raggedtwoe@raggedrightboxes@opt `\@raggedtwoe@raggedrightboxes@opt` is the code executed via `\DeclareOption`.

```
\newcommand*{\@raggedtwoe@raggedrightboxes@opt}{
```

First we check if `\@arrayparboxrestore` is unchanged.

```
  \CheckCommand*{\@arrayparboxrestore}{%
    \let\if@nobreak\iffalse
    \let\if@noskipsec\iffalse
    \let\par\@@par
    \let\-\@dischyph
    \let\'\@acci\let\`\@accii\let\=\@acciii
```

```
    \parindent\z@ \parskip\z@skip
    \everypar{}%
    \linewidth\hsize
    \@totalleftmargin\z@
    \leftskip\z@skip \rightskip\z@skip \@rightskip\z@skip
    \parfillskip\@flushglue
    \lineskip\normallineskip
    \lineskiplimit\normallineskiplimit
    \baselineskip\normalbaselineskip
    \sloppy}%
```

Then we redefine it by removing the setting of `\leftskip`, `\rightskip`, `\@rightskip` and `\parfillskip` and instead calling `\RaggedRight`.

```
  \renewcommand{\@arrayparboxrestore}{%
    \let\if@nobreak\iffalse
    \let\if@noskipsec\iffalse
    \let\par\@@par
    \let\-\@dischyph
    \let\'\@acci\let\`\@accii\let\=\@acciii
    \parskip\z@skip
    \everypar{}%
    \linewidth\hsize
    \@totalleftmargin\z@
    \RaggedRight
    \lineskip\normallineskip
    \baselineskip\normalbaselineskip
    \lineskiplimit\normallineskiplimit
    \sloppy}%
```

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

```
  \let\@raggedtwoe@raggedrightboxes@opt\relax
  }
```

Finally the declaration of the option.

```
\DeclareOption{raggedrightboxes}{\@raggedtwoe@raggedrightboxes@opt}
```

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.

```
\DeclareOption{footnotes}{%
  \@raggedtwoe@footmisctrue
  \PassOptionsToPackage{ragged}{footmisc}%
  }
```

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 `\@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
  }
```

\@raggedtwoe@document@opt `\@raggedtwoe@document@opt` is the code executed via `\DeclareOption`: Insert the code into `\AtBeginDocument`, execute the raggedrightboxes and footnotes options and self-destroy.

```
\newcommand{\@raggedtwoe@document@opt}{%
  \AtBeginDocument{\@raggedtwoe@abdhook}%
  \@raggedtwoe@raggedrightboxes@opt
  \@raggedtwoe@footmisctrue
  \let\@raggedtwoe@document@opt\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 everysel package for older kernels.

```
\providecommand\IfFormatAtLeastTF{\@ifl@t@r\fmtversion}
\IfFormatAtLeastTF{2021/01/05}{}{\RequirePackage{everysel}}
```

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).

```
\if@raggedtwoe@footmisc
  \RequirePackage{footmisc}[2002/08/28]
\fi
```

6.5 Allocations

`\CenteringLeftskip`
`\RaggedLeftLeftskip`
`\RaggedRightLeftskip`
`\CenteringRightskip`
`\RaggedLeftRightskip`
`\RaggedRightRightskip`
`\CenteringParfillskip`
`\RaggedLeftParfillskip`
`\RaggedRightParfillskip`
`\JustifyingParfillskip`
`\CenteringParindent`
`\RaggedLeftParindent`
`\RaggedRightParindent`
`\JustifyingParindent`

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
```

```
    \CenteringParfillskip\z@skip
    \RaggedLeftParfillskip\z@skip
    \RaggedRightParfillskip\@flushglue
    \CenteringParindent\z@
    \RaggedLeftParindent\z@
    \RaggedRightParindent\z@
\fi
\JustifyingParfillskip\@flushglue
\JustifyingParindent\parindent
```

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 *everytime* a font is loaded. Otherwise the settings for e. g. a proportional fonts would be in effect if you start `\RaggedRight` in `\rmfamily` and then switch to `\ttfamily`.

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

We do the setting with the help of everysel [8], which allows us to define code which is (hopefully) executed after every fontchange in a LATEX document.‡

`\if@raggedtwoe@spaceskip` `\if@raggedtwoe@spaceskip` signals the use of commands defined by ragged2e to the command inserted into `\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` `\@raggedtwoe@everyselectfont` is our code inserted into `\selectfont`.

```
\newcommand{\@raggedtwoe@everyselectfont@active}{%
  \if@raggedtwoe@spaceskip
```

If no command defined by ragged2e is in use, we do nothing. But if it is, we look at `\fontdimen3` to see if the current font is monospaced or not.

```
    \ifdim\fontdimen\thr@@\font=\z@\relax
```

If it is, we set `\spaceskip` to `0pt` so the interword space will be the one specified by the font designer – which is rigid anyway for monospaced fonts.

```
      \spaceskip\z@
    \else
```

‡It *is* executed after every `\selectfont`, so if you stay within NFSS and don't declare your fonts with commands like `\newfont` and then switch to them, it will work.

For proportional fonts we make the interword space rigid by setting `\spaceskip` to `\fontdimen2`.

```
      \spaceskip\fontdimen\tw@\font
    \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 everysel macro.

```
\newif\if@raggedtwoe@localhook
\IfFormatAtLeastTF{2021/01/05}
   {\@raggedtwoe@localhooktrue}
   {\EverySelectfont{\@raggedtwoe@everyselectfont}}
```

The hook is added globally but will be inactive when none of the ragged2e comands is used.

```
\let\@raggedtwoe@everyselectfont\relax
\if@raggedtwoe@localhook
  \AddToHook{selectfont}[ragged2e]{\@raggedtwoe@everyselectfont}%
\fi
%     \end{macrocode}
%  \end{macro}
%
%
% ^^A ----------------------------
%
%  \subsection{The commands}
%  ^^A
%  \begin{macro}{\@raggedtwoe@savedcr}
%  We save the definition of \cs{\textbackslash} in
%  \cs{@raggedtwoe@savedcr}.
%    \begin{macrocode}
\let\@raggedtwoe@savedcr\\
```

`\@raggedtwoe@saved@gnewline` We save the definition of `\@gnewline` in `\@raggedtwoe@saved@gnewline`.

```
\let\@raggedtwoe@saved@gnewline\@gnewline
```

`\@raggedtwoe@centercrNoindent` With the bugfix in version 3.2 we also take care of the indent which might be set when ragged2e's newline definition or `\@centercr` are triggered. This can be easily archieved using the ltpara hooks. This is not a final solution for this because on the long run one has to replace the paragraph break within `\@centercr` to match the content structure for tagging.

```
\def\@raggedtwoe@centercrNoindent{%
  \AddToHookNext{para/begin}{%
    \ifx\@raggedtwoe@everyselectfont\relax\else
      \OmitIndent
```

```
    \fi
  }%
   \@centercr%
}
```

`\@raggedtwoe@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
```

In formulas we adjusted the original code of `\g@newline` to a negative fill to match the alignment of text. Outside those we use `\@centercr` to remove the added parskip.

```
      \reserved@e {\reserved@f #1}\nobreak \hskip\z@ \break
    \else
      \reserved@e {\reserved@f #1}%
         \@raggedtwoe@centercrNoindent
    \fi
  \fi
  }
```

`\Centering` `\Centering` first lets `\\` = `\@centercr`, but only if `\\` has its original meaning, otherwise `\Center` would not work inside environments like 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.

```
\DeclareRobustCommand{\Centering}{%
   \if@raggedtwoe@localhook
    \let\@raggedtwoe@everyselectfont\@raggedtwoe@everyselectfont@active%
   \fi
   \ifx\\\@raggedtwoe@savedcr
      \let\\\@raggedtwoe@centercrNoindent
   \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}{%
   \if@raggedtwoe@localhook
        \let\@raggedtwoe@everyselectfont\@raggedtwoe@everyselectfont@active%
   \fi
   \ifx\\\@raggedtwoe@savedcr
      \let\\\@raggedtwoe@centercrNoindent
   \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}{%
   \if@raggedtwoe@localhook
    \let\@raggedtwoe@everyselectfont\@raggedtwoe@everyselectfont@active%
   \fi
   \ifx\\\@raggedtwoe@savedcr
      \let\\\@raggedtwoe@centercrNoindent
   \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 justyfied text.

```
\DeclareRobustCommand{\justifying}{%
   \if@raggedtwoe@localhook
    \let\@raggedtwoe@everyselectfont\@raggedtwoe@everyselectfont@active%
   \fi
   \let\\\@raggedtwoe@savedcr
   \let\@gnewline\@raggedtwoe@saved@gnewline
   \leftskip\z@
   \@rightskip\z@
   \rightskip\@rightskip
   \parfillskip\JustifyingParfillskip
   \parindent\JustifyingParindent
   \@raggedtwoe@spaceskipfalse
   \@raggedtwoe@everyselectfont
   }
```

6.8 The environments

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

```
\newenvironment{Center}{%
    \trivlist
    \Centering\item\relax
    }{%
    \endtrivlist
    }
\newenvironment{FlushLeft}{%
    \trivlist
    \RaggedRight\item\relax
    }{%
    \endtrivlist
    }
\newenvironment{FlushRight}{%
    \trivlist
    \RaggedLeft\item\relax
    }{%
    \endtrivlist
    }
```

`justify` (*env.*) justify is similar to the other environments: Start a trivlist and use `\justifying`.

```
\newenvironment{justify}{%
    \trivlist
    \justifying\item\relax
    }{%
    \endtrivlist
    }
```

6.9 Overloading the LaTeX-commands

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

```
\if@raggedtwoe@originalcommands
\else
    \let\LaTeXcentering\centering
    \let\LaTeXraggedleft\raggedleft
    \let\LaTeXraggedright\raggedright
    \let\centering\Centering
    \let\raggedleft\RaggedLeft
    \let\raggedright\RaggedRight
    \let\LaTeXcenter\center
    \let\endLaTeXcenter\endcenter
    \let\LaTeXflushleft\flushleft
    \let\endLaTeXflushleft\endflushleft
    \let\LaTeXflushright\flushright
```

```
    \let\endLaTeXflushright\endflushright
    \let\center\Center
    \let\endcenter\endCenter
    \let\flushleft\FlushLeft
    \let\endflushleft\endFlushLeft
    \let\flushright\FlushRight
    \let\endflushright\endFlushRight
\fi
```

⟨/package⟩

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.
Maurice Hansen reported and helped testing the fix for the bug adding an additional parskip fixed in version 3.2.

References

[1] Johannes Braams, David Carlisle, Alan Jeffrey, Leslie Lamport, Frank Mittelbach, Chris Rowley, and Rainer Schöpf. ltmiscen.dtx. Part of the LaTeX-distribution.

[2] Robin Fairbairns. `footmisc` — a portmanteau package for customising footnotes in LaTeX 2ε. https://mirror.ctan.org/macros/latex/contrib/footmisc/.

[3] James Kilfiger. https://ctan.org/tex-archive/obsolete/macros/latex/contrib/misc/raggedr.sty. LaTeX 2ε package.

[4] Donald E. Knuth. *The TeXBook*, volume A of *Computers & Typesetting*. Addison-Wesley, Reading, MA, USA, 1986.

[5] Leslie Lamport. *LaTeX: A Document Preparation System*. Addison-Wesley, Reading, MA, USA, first edition, 1986.

[6] Frank Mittelbach. The lthooks package. http://mirrors.ctan.org/macros/latex/base/lthooks-doc.pdf

[7] Frank Mittelbach and Rainer Schöpf. The file `cmfonts.fdd` for use with LaTeX 2ε. Part of the LaTeX-distribution.

[8] Martin Schröder. The obsolete everysel-package. http://mirrors.ctan.org/macros/latex/contrib/everysel/everysel.pdf. LaTeX 2ε package.

[9] Martin Schröder. The raggedri document option. Was in http://mirrors.ctan.org/tex-archive/macros/latex209/contrib/raggedright. LaTeX2.09 style, outdated.

[10] Laurent Siebenmann. `ragged.sty`. CTAN:tex-archive/macros/generic/ragged.sty. generic macro file for `plain` and 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.

Symbols

\' 39, 55
\- 38, 54
\= 39, 55
\@@par 37, 53
\@acci 39, 55
\@accii 39, 55
\@acciii 39, 55
\@arrayparboxrestore 34, 50
\@centercr 4, 182
\@dischyph 38, 54
\@flushglue 5, 10, 45, 114, 115, 117, 119, 122, 135, 140
\@gnewline . . . 175, 204, 220, 236, 250
\@ifl@t@r 94
\@ifundefined 75
\@nolnerr 186
\@plus 127, 128, 130, 132
\@raggedtwoe@abdhook 72, 79
\@raggedtwoe@centercrNoindent 176, 193, 202, 218, 234
\@raggedtwoe@document@opt . . 78, 84
\@raggedtwoe@everyselectfont 143, 178, 199, 211, 215, 227, 231, 243, 247, 257
\@raggedtwoe@everyselectfont@active 143, 199, 215, 231, 247
\@raggedtwoe@footmisctrue . . 69, 81
\@raggedtwoe@gnewline 184, 204, 220, 236
\@raggedtwoe@localhooktrue . . . 156
\@raggedtwoe@originalcommandsfalse 27, 28
\@raggedtwoe@originalcommandstrue 25, 26
\@raggedtwoe@originalparametersfalse 31, 32
\@raggedtwoe@originalparameterstrue 29, 30
\@raggedtwoe@raggedrightboxes@opt 33, 67, 80
\@raggedtwoe@saved@gnewline 175, 250
\@raggedtwoe@savedcr 170, 174, 201, 217, 233, 249
\@raggedtwoe@spaceskipfalse . . 256
\@raggedtwoe@spaceskiptrue 210, 226, 242
\@rightskip 5, 6, 44, 206, 207, 222, 223, 238, 239, 252, 253
\@tocrmarg 75
\@totalleftmargin 43, 59
\\ 4, 174, 201, 202, 217, 218, 233, 234, 249
\‘ 39, 55

A

\AddToHook 160
\AddToHookNext 177
\AtBeginDocument 79

B

\baselineskip 48, 62
\begin 170, 173
\break 190

C

\Center 297
Center (env.) 259
\center 291, 297
\Centering 197, 261, 288
\centering 285, 288
\CenteringLeftskip 99, 114, 127, 205
\CenteringParfillskip 99, 120, 133, 208
\CenteringParindent 99, 123, 136, 209
\CenteringRightskip 99, 117, 130, 206
\CheckCommand 34
\cs 171, 172
\CurrentOption 87

D

\DeclareOption 25, 26, 27, 28, 29, 30, 31, 32, 67, 68, 84, 85
\DeclareRobustCommand 2, 197, 213, 229, 245
\def 13, 17, 176

E

\edef 75
\else 88, 126, 147, 150, 178, 187, 191, 284
\end 162, 163
\endCenter 298
\endcenter 292, 298
\endFlushLeft 300
\endflushleft 294, 300
\endFlushRight 302
\endflushright 296, 302
\endLaTeXcenter 292
\endLaTeXflushleft 294
\endLaTeXflushright 296
\endtrivlist 263, 269, 275, 281
environments:
  Center 259
  FlushLeft 259
  FlushRight 259
  justify 277
\everypar 41, 57
\EverySelectfont 157
\ExecuteOptions 92

F

\fi 90, 98, 139, 149, 152, 161, 180, 194, 195, 200, 203, 216, 219, 232, 235, 248, 303
\finalhyphendemerits 7
\FlushLeft 299
FlushLeft (env.) 259
\flushleft 293, 299
\FlushRight 301
FlushRight (env.) 259
\flushright 295, 301
\fmtversion 94
\font 145, 148
\fontdimen 145, 148

H

\hsize 42, 58
\hskip 190

I

\if@nobreak 35, 51
\if@noskipsec 36, 52
\if@raggedtwoe@footmisc 24, 86, 96
\if@raggedtwoe@localhook 154, 159, 198, 214, 230, 246
\if@raggedtwoe@originalcommands 22, 283
\if@raggedtwoe@originalparameters 23, 113
\if@raggedtwoe@spaceskip . 142, 144
\ifdim 145
\iffalse 35, 36, 51, 52
\IfFormatAtLeastTF 94, 95, 155
\ifmmode 189
\ifvmode 185
\ifx 178, 201, 217, 233
\item 261, 267, 273, 279

J

justify (env.) 277
\justifying 245, 279
\JustifyingParfillskip 99, 140, 254
\JustifyingParindent . 99, 141, 255

L

\LaTeXcenter 291
\LaTeXcentering 285
\LaTeXflushleft 293
\LaTeXflushright 295
\LaTeXraggedleft 286

\LaTeXraggedright 287
\leftskip . . 8, 44, 205, 221, 237, 251
\let 4, 35, 36, 37, 38, 39, 51, 52, 53, 54, 55, 65, 76, 82, 158, 174, 175, 199, 202, 204, 215, 218, 220, 231, 234, 236, 247, 249, 250, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302
\lineskip 46, 61
\lineskiplimit 47, 63
\linewidth 42, 58

N

\newcommand 33, 72, 78, 143, 184
\newenvironment . 259, 265, 271, 277
\newif 22, 23, 24, 142, 154
\newlength 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112
\nobreak 190
\normalbaselineskip 48, 62
\normallineskip 46, 61
\normallineskiplimit 47, 63

O

\OmitIndent 179
\OptionNotUsed 89

P

\PackageInfo 73
\par 37, 53
\parfillskip . 45, 208, 224, 240, 254
\parindent 9, 40, 141, 209, 225, 241, 255
\parskip 40, 56
\PassOptionsToPackage 70, 87
\ProcessOptions 93
\providecommand 94

R

\RaggedLeft 213, 273, 289
\raggedleft 286, 289
\RaggedLeftLeftskip 99, 115, 128, 221
\RaggedLeftParfillskip 99, 121, 134, 224
\RaggedLeftParindent 99, 124, 137, 225
\RaggedLeftRightskip 99, 118, 131, 222
\RaggedRight 60, 73, 74, 229, 267, 290
\raggedright 3, 13, 287, 290
\RaggedRightLeftskip 99, 116, 129, 237
\RaggedRightParfillskip 99, 122, 135, 240
\RaggedRightParindent 99, 125, 138, 241
\RaggedRightRightskip 99, 119, 132, 238
\relax 16, 19, 65, 76, 82, 93, 145, 158, 178, 261, 267, 273, 279
\renewcommand 50
\RequirePackage 95, 97
\reserved@e 190, 192
\reserved@f 190, 192
\rightskip 6, 14, 19, 44, 207, 223, 239, 253

S

\sloppy 49, 64
\spaceskip 15, 146, 148, 151
\string 73
\subsection 168

T

\textbackslash 171
\thr@@ 145
\trivlist 260, 266, 272, 278
\tt 18
\ttraggedright 17
\tw@ 127, 128, 130, 132, 148

U

\unskip 188

X

\xspaceskip 16

Z

\z@ 7, 9, 14, 19, 40, 43, 59, 123, 124, 125, 127, 128, 130, 132, 136, 137, 138, 145, 146, 151, 190, 251, 252
\z@skip 8, 40, 44, 56, 116, 118, 120, 121, 129, 131, 133, 134

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
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
\@raggedtwoe@abdhook: New macro 9
\@raggedtwoe@document@opt: New macro 9
\@raggedtwoe@everyselectfont: Completely redesigned and removed \RaggedSpaceskip and \RaggedXSpaceskip 11
\@raggedtwoe@gnewline: New macro 13
\@raggedtwoe@raggedrightboxes@opt: New macro 7
\@raggedtwoe@saved@gnewline: New macro 12
\Centering: Call \@raggedtwoe@everyselectfont and switch \@gnewline 13
\if@raggedtwoe@footmisc: New macro 7
justify: New environment 15
\justifying: New macro 14
\JustifyingParfillskip: New macro 10
\JustifyingParindent: New macro 10
\RaggedLeft: Call \@raggedtwoe@everyselectfont and switch \@gnewline 14
\RaggedRight: Call \@raggedtwoe@everyselectfont and switch \@gnewline 14
v2.01
\@raggedtwoe@everyselectfont: Removed the setting of \xspaceskip . 11
v2.02
General: Bugfix: \if@raggedtwoe@footmisctrue → \@raggedtwoe@footmisctrue 8
Removed \setlength 6
Use \@flushglue 6
\@raggedtwoe@gnewline: Bugfix: \@nolerr → \@nolnerr 13
FlushRight: Use \trivlist ... \endtrivlist instead of \begin{trivlist} ... \end{trivlist} 15
justify: Use \trivlist ... \endtrivlist instead of \begin{trivlist} ... \end{trivlist} 15

v2.03
General: Bugfix: footnotes was actually raggedrightboxes 8
v2.04
General: Initialize \JustifyingParindent with \parindent 10
Insert missing \ 10
Save more commands 15
\@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
v2.1
General: bugfix: \Flushleft instead of \FlushLeft (found by Berend Hasselman) 15
bugfix: Load footmisc directly and not via \AtEndOfPackage (bug found by Axel Sommerfeldt) 10
document that document needs footmisc 6
\@raggedtwoe@abdhook: bugfix: Use \@tocrmarg only if it's defined 9
\@raggedtwoe@gnewline: Bugfix: handle math 13
v2.2
General: Change maintenance status of package 2
Move to git/gitlab, use svninfo2 1
Require a new version of LaTeX(2017/03/29) 1
\@raggedtwoe@raggedrightboxes@opt: Definition of \@arrayparboxrestore has changed 7
v3.0
General: Change maintenance status 2
document everysel is obsolete 6
v3.00
General: Remove the everysel package if kernel hooks are available 9
\@raggedtwoe@everyselectfont: Use kernel hook if available 12
v3.1
General: Use the updated definition of \raggedright 2
\Centering: Robustify the user macros (Thanks to Markus Kohm for the hint) 13
\justifying: Robustify the user macros (Thanks to Markus Kohm for the hint) 14
\RaggedLeft: Robustify the user macros (Thanks to Markus Kohm for the hint) 14
\RaggedRight: Robustify the user macros (Thanks to Markus Kohm for the hint) 14
v3.2
\@raggedtwoe@centercrNoindent: New macro 12
\@raggedtwoe@gnewline: Fix bug adding parskip with newline 13
v3.3
\@raggedtwoe@everyselectfont: Apply hook only when a ragged2e command is active. 12
\Centering: Apply hook only when a ragged2e command is active 13
\justifying: Apply hook only when a ragged2e command is active 14
\RaggedLeft: Apply hook only when a ragged2e command is active 14
\RaggedRight: Apply hook only when a ragged2e command is active . . . 14
v3.4
\Centering: Bugfix: remove spurious space 13

`\justifying`: Bugfix: remove spurious space 14
`\RaggedLeft`: Bugfix: remove spurious space 14
`\RaggedRight`: Bugfix: remove spurious space 14
v3.5
`\@raggedtwoe@everyselectfont`: Add mechanism to deactivate `\raggedtwoe@everyselectfont` 11
Globally add inactive hook 12
`\Centering`: Bugfix: Only activate the global hook 13
`\justifying`: Bugfix: Only activate the global hook 14
`\RaggedLeft`: Bugfix: Only activate the global hook 14
`\RaggedRight`: Bugfix: Only activate the global hook 14
v3.6
`\@raggedtwoe@centercrNoindent`: Only omit indent if inside a group where a ragged2e command is active 13