

# The ragged2e-package\*

Martin Schröder

Unterstraße 31

47051 Duisburg

Germany

[martin@oneiros.de](mailto:martin@oneiros.de)

<https://gitlab.com/hagenbuch/ragged2e>

2019/07/28

## 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` [8] by the same author.

## Contents

<b>1</b>	<b>Status of this package</b>	<b>2</b>
<b>2</b>	<b>The problem</b>	<b>2</b>
<b>3</b>	<b>Old “solutions”</b>	<b>2</b>
3.1	<code>L<sup>A</sup>T<sub>E</sub>X</code> . . . . .	2
3.2	<code>plain</code> . . . . .	3
<b>4</b>	<b>Our solution</b>	<b>3</b>
4.1	The macros . . . . .	3
4.2	The parameters . . . . .	4
4.3	The environments . . . . .	5
<b>5</b>	<b>Options</b>	<b>5</b>
<b>6</b>	<b>Required packages</b>	<b>6</b>
<b>7</b>	<b>The implementation</b>	<b>6</b>
7.1	Initial Code . . . . .	6
7.2	Declaration of options . . . . .	7
7.2.1	<code>originalcommands</code> option . . . . .	7
7.2.2	<code>originalparameters</code> option . . . . .	7
7.2.3	<code>raggedrightboxes</code> option . . . . .	7

---

\*The version number of this file is v2.2, revision #(None), last revised (None).

7.2.4	footnotes option	8
7.2.5	document option	8
7.2.6	Other options	9
7.3	Executing options	9
7.4	Loading packages	9
7.5	Allocations	10
7.6	Initializations	10
7.7	Distinguishing between monospaced and proportional fonts	11
7.8	The commands	12
7.9	The environments	14
7.10	Overloading the L <sup>A</sup> T <sub>E</sub> X-commands	14

## 8 Acknowledgements 15

### 1 Status of this package

This package is *unmaintained*.

I will not actively work on this package anymore. I *may* merge merge requests to the repository on Gitlab (<https://gitlab.com/hagenbuch/ragged2e>). If you want to become the maintainer of this package, contact me and I may hand over the repository to you. Or fork the package on Gitlab.

I will *not* upload new versions to CTAN.

### 2 The problem

L<sup>A</sup>T<sub>E</sub>X 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:

<code>\raggedright:</code>	<code>\RaggedRight:</code>
“The L <sup>A</sup> T <sub>E</sub> X document preparation system is a special version of Donald Knuth’s T <sub>E</sub> X program. T <sub>E</sub> X is a sophisticated program designed to produce high-quality typesetting, especially for mathematical text.” [5, p. xiii]	“The L <sup>A</sup> T <sub>E</sub> X document preparation system is a special version of Donald Knuth’s T <sub>E</sub> X program. T <sub>E</sub> X is a sophisticated program designed to produce high-quality typesetting, especially for mathematical text.” [5, p. xiii]

### 3 Old “solutions”

#### 3.1 L<sup>A</sup>T<sub>E</sub>X

L<sup>A</sup>T<sub>E</sub>X defines e. g. `\raggedright` as follows:

```

1 <*\latex>
2 \def\raggedright{%
3   \let\=\@centercr
4   \@rightskip\@flushglue
5   \rightskip\@rightskip
6   \leftskip\z@
7   \parindent\z@}

```

Initially, `\@flushglue` is defined as

```

8 \@flushglue = 0pt plus 1fil
9 </\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 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.”

### 3.2 plain

`plain` T<sub>E</sub>X defines a special version of `\raggedright`, which operates the way Knuth describes it; but which can not be used with L<sup>A</sup>T<sub>E</sub>X, because L<sup>A</sup>T<sub>E</sub>X redefines `\raggedright`.

```
10 <*plain>
11 \def\raggedright{%
12   \rightskip\z@ plus2em
13   \spaceskip.3333em
14   \xspaceskip.5em\relax}
```

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

```
15 \def\tt\raggedright{%
16   \tt
17   \rightskip\z@ plus2em\relax}
18 </plain>
```

## 4 Our solution

Since the `plain` solution can not be used with L<sup>A</sup>T<sub>E</sub>X, we have to redefine it and make it possible to configure it for personal preferences.

### 4.1 The macros

`\Centering` `\RaggedLeft` `\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.

The new commands `\Centering`, `\RaggedLeft`, and `\RaggedRight` are fully compatible with their counterparts in L<sup>A</sup>T<sub>E</sub>X, but implement the `plain` solution and can be easily configured using the following parameters:

---

\*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.

## 4.2 The parameters

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

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

```
\setlength{\RaggedRightRightskip}{Opt plus 1em}
```

sets `\RaggedRightRightskip` to `Opt plus 1em`.

`\CenteringLeftskip`  
`\RaggedLeftLeftskip`  
`\RaggedRightLeftskip` These are the `\leftskips` inserted by `\Centering`, `\RaggedLeft`, and `\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 `Opt 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 `Opt 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 `Opt 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 `Opt`, since indented lines in ragged text do not look good.

The parameters have the following initial setting:

Parameter	L <sup>A</sup> T <sub>E</sub> X setting	ragged2e setting <sup>†</sup>
<code>\CenteringLeftskip</code>	Opt plus 1fil	Opt plus 2em
<code>\RaggedLeftLeftskip</code>	Opt plus 1fil	Opt plus 2em
<code>\RaggedRightLeftskip</code>	Opt plus Opt minus Opt	Opt plus Opt minus Opt
<code>\CenteringRightskip</code>	Opt plus 1fil	Opt plus 2em
<code>\RaggedLeftRightskip</code>	Opt plus Opt minus Opt	Opt plus Opt minus Opt
<code>\RaggedRightRightskip</code>	Opt plus 1fil	Opt plus 2em
<code>\CenteringParfillskip</code>	Opt plus Opt minus Opt	Opt plus Opt minus Opt
<code>\RaggedLeftParfillskip</code>	Opt plus Opt minus Opt	Opt plus Opt minus Opt
<code>\RaggedRightParfillskip</code>	Opt plus 1fil	Opt plus 1fil
<code>\CenteringParindent</code>	Opt	Opt
<code>\RaggedLeftParindent</code>	Opt	Opt
<code>\RaggedRightParindent</code>	Opt	Opt
<code>\JustifyingParfillskip</code>		Opt plus 1fil
<code>\JustifyingParindent</code>		<code>\parindent</code>

### 4.3 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`.

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}
```

## 5 Options

This package has the following options:

`originalcommands` The L<sup>A</sup>T<sub>E</sub>X-commands `\centering`, `\raggedleft`, and `\raggedright` and the L<sup>A</sup>T<sub>E</sub>X-environments `center`, `flushleft`, and `flushright` remain unchanged.

It is the default.

<sup>†</sup>For proportional and monospaced fonts.

**newcommands** The L<sup>A</sup>T<sub>E</sub>X-commands `\centering`, `\raggedleft`, and `\raggedright` and the L<sup>A</sup>T<sub>E</sub>X-environments `center`, `flushleft`, and `flushright` are set equal to their counterparts defined by `ragged2e`. Thus `\raggedright` invokes `\RaggedRight`. The original commands can be accessed under the Names `\LaTeX $\langle original name \rangle$` , e. g. `\LaTeXraggedright`.

**originalparameters** The parameters used by the commands implemented by `ragged2e` are initialized with the default settings used by L<sup>A</sup>T<sub>E</sub>X.

**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 `\parboxes`, `minipages`, `\marginpars` 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.

## 6 Required packages

This package requires the following packages:

`everysel` [7] It is used to distinguish between monospaced and proportional fonts.

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

## 7 The implementation

19 `\*package)`

### 7.1 Initial Code

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

20 `\newif\if@raggedtwoe@originalcommands`

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

21 `\newif\if@raggedtwoe@originalparameters`

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

## 7.2 Declaration of options

### 7.2.1 originalcommands option

The `originalcommands` and `newcommands` options control the meaning of the L<sup>A</sup>T<sub>E</sub>X-commands for ragged text: If `newcommands` is used the L<sup>A</sup>T<sub>E</sub>X-commands are set equal to the commands defined by `ragged2e`.

```
23 \DeclareOption{OriginalCommands}{\@raggedtwoe@originalcommandstrue}
24 \DeclareOption{originalcommands}{\@raggedtwoe@originalcommandstrue}
25 \DeclareOption{NewCommands}{\@raggedtwoe@originalcommandsfalse}
26 \DeclareOption{newcommands}{\@raggedtwoe@originalcommandsfalse}
```

### 7.2.2 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`.

```
27 \DeclareOption{OriginalParameters}{\@raggedtwoe@originalparameterstrue}
28 \DeclareOption{originalparameters}{\@raggedtwoe@originalparameterstrue}
29 \DeclareOption{NewParameters}{\@raggedtwoe@originalparametersfalse}
30 \DeclareOption{newparameters}{\@raggedtwoe@originalparametersfalse}
```

### 7.2.3 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.
```

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

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

```
32 \CheckCommand*{\@arrayparboxrestore}{%
33 \let\if@nobreak\iffalse
34 \let\if@noskipsec\iffalse
35 \let\par\@par
36 \let-\@dischyph
37 \let'\@acci\let'\@accii\let\=\@acciii
38 \parindent\z@ \parskip\z@skip
39 \everypar{}%
40 \linewidth\hsize
41 \@totalleftmargin\z@
42 \leftskip\z@skip \rightskip\z@skip \@rightskip\z@skip
```

```

43 \parfillskip\@flushglue
44 \lineskip\normallineskip
45 \lineskiplimit\normallineskiplimit
46 \baselineskip\normalbaselineskip
47 \sloppy}%

```

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

```

48 \renewcommand{\arrayparboxrestore}{%
49 \let\if@nobreak\iffalse
50 \let\if@noskipsec\iffalse
51 \let\par\@par
52 \let\-\@dischph
53 \let'\@acci\let'\@accii\let\=\@acciii
54 \parskip\z@skip
55 \everypar{}%
56 \linewidth\hsize
57 \@totalleftmargin\z@
58 \RaggedRight
59 \lineskip\normallineskip
60 \baselineskip\normalbaselineskip
61 \lineskiplimit\normallineskiplimit
62 \sloppy}%

```

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

```

63 \let\@raggedtwoe@raggedrightboxes@opt\relax
64 }

```

Finally the declaration of the option.

```

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

```

#### 7.2.4 footnotes option

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

```

66 \DeclareOption{footnotes}{%
67 \@raggedtwoe@footmisctrue
68 \PassOptionsToPackage{ragged}{footmisc}%
69 }

```

#### 7.2.5 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.

```
70 \newcommand{\@raggedtwo@abdhook}{%
71   \PackageInfo{ragged2e}{ABD: executing \string\RaggedRight}%
72   \RaggedRight
73   \@ifundefined{@tocrmarg}{\edef\@tocrmarg{\@tocrmarg plus 2em}}%
74   \let\@raggedtwo@abdhook\relax
75 }
```

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

```
76 \newcommand{\@raggedtwo@document@opt}{%
77   \AtBeginDocument{\@raggedtwo@abdhook}%
78   \@raggedtwo@raggedrightboxes@opt
79   \@raggedtwo@footmisctrue
80   \let\@raggedtwo@document@opt\relax
81 }
```

Finally the declaration of the option.

```
82 \DeclareOption{document}{\@raggedtwo@document@opt}
```

### 7.2.6 Other options

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

```
83 \DeclareOption*{%
84   \if@raggedtwo@footmisc
85     \PassOptionsToPackage{\CurrentOption}{footmisc}%
86   \else
87     \OptionNotUsed
88   \fi
89 }
```

## 7.3 Executing options

The default options are `originalcommands` and `newparameters`.

```
90 \ExecuteOptions{originalcommands,newparameters}
91 \ProcessOptions\relax
```

## 7.4 Loading packages

We need the `everysel` package.

```
92 \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).

```
93 \if@raggedtwo@footmisc
94   \RequirePackage{footmisc}[2002/08/28]
95 \fi
```

## 7.5 Allocations

```
\CenteringLeftskip First we allocate the parameters
\RaggedLeftLeftskip 96 \newlength{\CenteringLeftskip}
\RaggedRightLeftskip 97 \newlength{\RaggedLeftLeftskip}
\CenteringRightskip 98 \newlength{\RaggedRightLeftskip}
\RaggedLeftRightskip 99 \newlength{\CenteringRightskip}
\RaggedRightRightskip 100 \newlength{\RaggedLeftRightskip}
\CenteringParfillskip 101 \newlength{\RaggedRightRightskip}
\RaggedLeftParfillskip 102 \newlength{\CenteringParfillskip}
\RaggedRightParfillskip 103 \newlength{\RaggedLeftParfillskip}
\JustifyingParfillskip 104 \newlength{\RaggedRightParfillskip}
\CenteringParindent 105 \newlength{\JustifyingParfillskip}
\RaggedLeftParindent 106 \newlength{\CenteringParindent}
\RaggedRightParindent 107 \newlength{\RaggedLeftParindent}
\JustifyingParindent 108 \newlength{\RaggedRightParindent}
109 \newlength{\JustifyingParindent}
```

## 7.6 Initializations

Depending on `\if@raggedtwo@originalparameters` we initialize the parameters with the values  $\LaTeX$  uses for its own commands or with our new parameters.

```
110 \if@raggedtwo@originalparameters
111   \CenteringLeftskip\@flushglue
112   \RaggedLeftLeftskip\@flushglue
113   \RaggedRightLeftskip\z@skip
114   \CenteringRightskip\@flushglue
115   \RaggedLeftRightskip\z@skip
116   \RaggedRightRightskip\@flushglue
117   \CenteringParfillskip\z@skip
118   \RaggedLeftParfillskip\z@skip
119   \RaggedRightParfillskip\@flushglue
120   \CenteringParindent\z@
121   \RaggedLeftParindent\z@
122   \RaggedRightParindent\z@
123 \else
124   \CenteringLeftskip\z@\@plus\tw@ em
125   \RaggedLeftLeftskip\z@\@plus\tw@ em
126   \RaggedRightLeftskip\z@skip
127   \CenteringRightskip\z@\@plus\tw@ em
128   \RaggedLeftRightskip\z@skip
129   \RaggedRightRightskip\z@\@plus\tw@ em
```

```

130 \CenteringParfillskip\z@skip
131 \RaggedLeftParfillskip\z@skip
132 \RaggedRightParfillskip\@flushglue
133 \CenteringParindent\z@
134 \RaggedLeftParindent\z@
135 \RaggedRightParindent\z@
136 \fi
137 \JustifyingParfillskip\@flushglue
138 \JustifyingParindent\parindent

```

## 7.7 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.  $\text{\TeX}$ '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` [7], which allows us to define code which is (hopefully) executed after every fontchange in a  $\text{\LaTeX}$  document.<sup>‡</sup>

```

\if@raggedtwo@spaceskip \if@raggedtwo@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  $\text{\TeX}$  when the scope of them
ends.

```

```

139 \newif\if@raggedtwo@spaceskip

```

```

\@raggedtwo@everyselectfont \@raggedtwo@everyselectfont is our code inserted into \selectfont.

```

```

140 \newcommand{\@raggedtwo@everyselectfont}{%
141 \if@raggedtwo@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.

```

142 \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.

```

143 \spaceskip\z@
144 \else

```

---

<sup>‡</sup>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`.

```
145     \spaceskip\fontdimen\tw@\font
146     \fi
```

We have to reset the interword space if we are not active.

```
147     \else
148     \spaceskip\z@
149     \fi
150   }
151 \EverySelectfont{\@raggedtwoe@everyselectfont}
```

## 7.8 The commands

`\@raggedtwoe@savedcr` We save the definition of `\` in `\@raggedtwoe@savedcr`.

```
152 \let\@raggedtwoe@savedcr\
```

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

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

`\@raggedtwoe@gnewline` The following definition of a `\@gnewline` used by the ragged commands was suggested by Markus Kohm:

```
154 \newcommand*{\@raggedtwoe@gnewline}[1]{%
155   \ifvmode
156     \nolnerr
157   \else
158     \unskip
159     \ifmmode
```

In formulas we use the original code of `\@gnewline`.

```
160     \reserved@e {\reserved@f #1}\nobreak \hfil \break
161   \else
162     \reserved@e {\reserved@f #1}{\parskip\z@\par}%
163   \fi
164 \fi
165 }
```

`\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  $\text{\LaTeX}$  and  $\text{\TeX}$ -parameters are set. `\@rightskip` is  $\text{\LaTeX}$ '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.

```

166 \newcommand{\Centering}{%
167   \ifx\\ \@raggedtwo@savedcr
168     \let\\ \@centercr
169   \fi
170   \let\@gnewline\@raggedtwo@gnewline
171   \leftskip\CenteringLeftskip
172   \@rightskip\CenteringRightskip
173   \rightskip\@rightskip
174   \parfillskip\CenteringParfillskip
175   \parindent\CenteringParindent
176   \@raggedtwo@spaceskiptrue
177   \@raggedtwo@everyselectfont
178 }

```

`\RaggedLeft` `\RaggedLeft` is like `\Centering`; it only uses other parameters.

```

179 \newcommand{\RaggedLeft}{%
180   \ifx\\ \@raggedtwo@savedcr
181     \let\\ \@centercr
182   \fi
183   \let\@gnewline\@raggedtwo@gnewline
184   \leftskip\RaggedLeftLeftskip
185   \@rightskip\RaggedLeftRightskip
186   \rightskip\@rightskip
187   \parfillskip\RaggedLeftParfillskip
188   \parindent\RaggedLeftParindent
189   \@raggedtwo@spaceskiptrue
190   \@raggedtwo@everyselectfont
191 }

```

`\RaggedRight` `\RaggedRight` is like `\Centering`; it only uses other parameters.

```

192 \newcommand{\RaggedRight}{%
193   \ifx\\ \@raggedtwo@savedcr
194     \let\\ \@centercr
195   \fi
196   \let\@gnewline\@raggedtwo@gnewline
197   \leftskip\RaggedRightLeftskip
198   \@rightskip\RaggedRightRightskip
199   \rightskip\@rightskip
200   \parfillskip\RaggedRightParfillskip
201   \parindent\RaggedRightParindent
202   \@raggedtwo@spaceskiptrue
203   \@raggedtwo@everyselectfont
204 }

```

`\justifying` `\justifying` switches back to the defaults used by L<sup>A</sup>T<sub>E</sub>X for typesetting justified text.

```

205 \newcommand{\justifying}{%
206   \let\\ \@raggedtwo@savedcr
207   \let\@gnewline\@raggedtwo@saved@gnewline
208   \leftskip\z@
209   \@rightskip\z@

```

```

210 \rightskip\@rightskip
211 \parfillskip\JustifyingParfillskip
212 \parindent\JustifyingParindent
213 \@raggedtwoe@spaceskipfalse
214 \@raggedtwoe@everyselectfont
215 }

```

## 7.9 The environments

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

```

216 \newenvironment{Center}{%
217 \trivlist
218 \Centering\item\relax
219 }{%
220 \endtrivlist
221 }
222 \newenvironment{FlushLeft}{%
223 \trivlist
224 \RaggedRight\item\relax
225 }{%
226 \endtrivlist
227 }
228 \newenvironment{FlushRight}{%
229 \trivlist
230 \RaggedLeft\item\relax
231 }{%
232 \endtrivlist
233 }

```

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

```

234 \newenvironment{justify}{%
235 \trivlist
236 \justifying\item\relax
237 }{%
238 \endtrivlist
239 }

```

## 7.10 Overloading the $\LaTeX$ -commands

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

```

240 \if@raggedtwoe@originalcommands
241 \else
242 \let\LaTeXcentering\centering
243 \let\LaTeXraggedleft\raggedleft
244 \let\LaTeXraggedright\raggedright
245 \let\centering\Centering

```

```

246 \let\raggedleft\RaggedLeft
247 \let\raggedright\RaggedRight
248 \let\LaTeXcenter\center
249 \let\endLaTeXcenter\endcenter
250 \let\LaTeXflushleft\flushleft
251 \let\endLaTeXflushleft\endflushleft
252 \let\LaTeXflushright\flushright
253 \let\endLaTeXflushright\endflushright
254 \let\center\Center
255 \let\endcenter\endCenter
256 \let\flushleft\FlushLeft
257 \let\endflushleft\endFlushLeft
258 \let\flushright\FlushRight
259 \let\endflushright\endFlushRight
260 \fi
261 </package>

```

## 8 Acknowledgements

A first version of this package for L<sup>A</sup>T<sub>E</sub>X 2.09 was named `raggedri` [8]. Laurent Siebenmann ([lcs@topo.math.u-psud.fr](mailto:lcs@topo.math.u-psud.fr)) with his style `ragged.sty` [9] 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](mailto:mapdn@csv.warwick.ac.uk)).

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

Markus Kohm ([markus.kohm@gmx.de](mailto:markus.kohm@gmx.de)) provided the code for `\@gnewline`.

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

Rolf Niepraschk ([Rolf.Niepraschk@gmx.de](mailto: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](mailto:jafirth@ncsu.edu)) provided the final push for version 2.2.

## References

- [1] Johannes Braams, David Carlisle, Alan Jeffrey, Leslie Lamport, Frank Mittelbach, Chris Rowley, and Rainer Schöpf. `ltxmisc.dtx`. Part of the L<sup>A</sup>T<sub>E</sub>X-distribution.
- [2] Robin Fairbairns. `footmisc` — a portmanteau package for customising footnotes in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. [CTAN:tex-archive/macros/latex/contrib/supported/footmisc/footmisc.dtx](http://CTAN:tex-archive/macros/latex/contrib/supported/footmisc/footmisc.dtx).
- [3] James Kilfinger. [CTAN:tex-archive/macros/latex/contrib/other/misc/raggedr.sty](http://CTAN:tex-archive/macros/latex/contrib/other/misc/raggedr.sty). L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> package.

- [4] Donald E. Knuth. *The T<sub>E</sub>XBook*, volume A of *Computers & Typesetting*. Addison-Wesley, Reading, MA, USA, 1986.
- [5] Leslie Lamport. *L<sup>A</sup>T<sub>E</sub>X: A Document Preparation System*. Addison-Wesley, Reading, MA, USA, first edition, 1986.
- [6] Frank Mittelbach and Rainer Schöpf. The file `cmfonts.fdd` for use with L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. Part of the L<sup>A</sup>T<sub>E</sub>X-distribution.
- [7] Martin Schröder. The `everysel`-package. [CTAN:tex-archive/macros/latex/contrib/supported/ms/everysel.dtx](#). L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> package.
- [8] Martin Schröder. The `raggedri` document option. Was in [CTAN:tex-archive/macros/latex209/contrib/raggedright](#). L<sup>A</sup>T<sub>E</sub>X 2.09 style, outdated.
- [9] Laurent Siebenmann. `ragged.sty`. [CTAN:tex-archive/macros/generic/ragged.sty](#). generic macro file for plain and L<sup>A</sup>T<sub>E</sub>X.

## 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	
<code>\'</code> .....	37, 53
<code>\-</code> .....	36, 52
<code>\=</code> .....	37, 53
<code>\@par</code> .....	35, 51
<code>\@acci</code> .....	37, 53
<code>\@accii</code> .....	37, 53
<code>\@acciii</code> .....	37, 53
<code>\@arrayparboxrestore</code> .....	32, 48
<code>\@centercr</code> .....	3, 168, 181, 194
<code>\@dischyph</code> .....	36, 52
<code>\@flushglue</code> .....	4, 8, 43, 111, 112, 114, 116, 119, 132, 137
<code>\@gnewline</code> ...	153, 170, 183, 196, 207
<code>\@ifundefined</code> .....	73
<code>\@nolnerr</code> .....	156
<code>\@plus</code> .....	124, 125, 127, 129
<code>\@raggedtwoe@abdhook</code> .....	70, 77
<code>\@raggedtwoe@document@opt</code> ..	76, 82
<code>\@raggedtwoe@everyselectfont</code> .	..... 140, 177, 190, 203, 214
<code>\@raggedtwoe@footmisctrue</code> ..	67, 79
<code>\@raggedtwoe@gnewline</code> .....	..... 154, 170, 183, 196
<code>\@raggedtwoe@originalcommandsfalse</code>	..... 25, 26
<code>\@raggedtwoe@originalcommandstrue</code>	..... 23, 24
<code>\@raggedtwoe@originalparametersfalse</code>	..... 29, 30
<code>\@raggedtwoe@originalparameterstrue</code>	..... 27, 28
<code>\@raggedtwoe@raggedrightboxes@opt</code>	..... 31, 65, 78
<code>\@raggedtwoe@samed@gnewline</code> ..	..... 153, 207
<code>\@raggedtwoe@samed@cr</code> .....	..... 152, 167, 180, 193, 206
<code>\@raggedtwoe@spaceskipfalse</code> ..	213
<code>\@raggedtwoe@spaceskiptrue</code> ...	..... 176, 189, 202
<code>\@rightskip</code> .....	4, 5, 42, 172, 173, 185, 186, 198, 199, 209, 210
<code>\@tocmarg</code> .....	73
<code>\@totalleftmargin</code> .....	41, 57
<code>\@</code> .....	3, 152, 167, 168, 180, 181, 193, 194, 206
<code>\'</code> .....	37, 53
<b>A</b>	
<code>\AtBeginDocument</code> .....	77



- B**
- `\baselineskip` . . . . . 46, 60
  - `\break` . . . . . 160
- C**
- `\Center` . . . . . 254
  - Center (environment) . . . . . 5, [216](#)
  - `\center` . . . . . 248, 254
  - `\Centering` . . . . . 3, [166](#), 218, 245
  - `\centering` . . . . . 242, 245
  - `\CenteringLeftskip` . . . . .
  - . . . . . 4, [96](#), 111, 124, 171
  - `\CenteringParfillskip` . . . . .
  - . . . . . 4, [96](#), 117, 130, 174
  - `\CenteringParindent` . . . . .
  - . . . . . 4, [96](#), 120, 133, 175
  - `\CenteringRightskip` . . . . .
  - . . . . . 4, [96](#), 114, 127, 172
  - `\CheckCommand` . . . . . 32
  - `\CurrentOption` . . . . . 85
- D**
- `\DeclareOption` . . . . . 23, 24, 25,
  - 26, 27, 28, 29, 30, 65, 66, 82, 83
  - `\def` . . . . . 2, 11, 15
- E**
- `\edef` . . . . . 73
  - `\else` 86, 123, 144, 147, 157, 161, 241
  - `\endCenter` . . . . . 255
  - `\endcenter` . . . . . 249, 255
  - `\endFlushLeft` . . . . . 257
  - `\endflushleft` . . . . . 251, 257
  - `\endFlushRight` . . . . . 259
  - `\endflushright` . . . . . 253, 259
  - `\endLaTeXcenter` . . . . . 249
  - `\endLaTeXflushleft` . . . . . 251
  - `\endLaTeXflushright` . . . . . 253
  - `\endtrivlist` . . . . . 220, 226, 232, 238
  - environments:
  - Center . . . . . 5, [216](#)
  - FlushLeft . . . . . 5, [216](#)
  - FlushRight . . . . . 5, [216](#)
  - justify . . . . . 5, [234](#)
  - `\everypar` . . . . . 39, 55
  - `\EverySelectfont` . . . . . 151
  - `\ExecuteOptions` . . . . . 90
- F**
- `\fi` . . . . . 88, 95, 136, 146,
  - 149, 163, 164, 169, 182, 195, 260
  - `\FlushLeft` . . . . . 256
  - FlushLeft (environment) . . . . . 5, [216](#)
  - `\flushleft` . . . . . 250, [256](#)
  - `\FlushRight` . . . . . 258
  - FlushRight (environment) . . . . . 5, [216](#)
  - `\flushright` . . . . . 252, 258
  - `\font` . . . . . 142, 145
  - `\fontdimen` . . . . . 142, 145
- H**
- `\hfil` . . . . . 160
  - `\hsize` . . . . . 40, 56
- I**
- `\if@nobreak` . . . . . 33, 49
  - `\if@noskipsec` . . . . . 34, 50
  - `\if@raggedtwo@footmisc` [22](#), 84, 93
  - `\if@raggedtwo@originalcommands`
  - . . . . . [20](#), 240
  - `\if@raggedtwo@originalparameters`
  - . . . . . [21](#), 110
  - `\if@raggedtwo@spaceskip` . [139](#), 141
  - `\ifdim` . . . . . 142
  - `\iffalse` . . . . . 33, 34, 49, 50
  - `\ifmode` . . . . . 159
  - `\ifvmode` . . . . . 155
  - `\ifx` . . . . . 167, 180, 193
  - `\item` . . . . . 218, 224, 230, 236
- J**
- justify (environment) . . . . . 5, [234](#)
  - `\justifying` . . . . . 3, [205](#), 236
  - `\JustifyingParfillskip` . . . . .
  - . . . . . 4, [96](#), 137, 211
  - `\JustifyingParindent` 4, [96](#), 138, 212
- L**
- `\LaTeXcenter` . . . . . 248
  - `\LaTeXcentering` . . . . . 242
  - `\LaTeXflushleft` . . . . . 250
  - `\LaTeXflushright` . . . . . 252
  - `\LaTeXraggedleft` . . . . . 243
  - `\LaTeXraggedright` . . . . . 244
  - `\leftskip` . . . . . 6, 42, 171, 184, 197, 208
  - `\let` . . . . . 3, 33,
  - 34, 35, 36, 37, 49, 50, 51, 52,
  - 53, 63, 74, 80, 152, 153, 168,
  - 170, 181, 183, 194, 196, 206,
  - 207, 242, 243, 244, 245, 246,
  - 247, 248, 249, 250, 251, 252,
  - 253, 254, 255, 256, 257, 258, 259
  - `\lineskip` . . . . . 44, 59
  - `\lineskiplimit` . . . . . 45, 61
  - `\linewidth` . . . . . 40, 56
- N**
- `\newcommand` . . . . . 31, 70,
  - 76, 140, 154, 166, 179, 192, 205

<code>\newenvironment</code> . . . . .	216, 222, 228, 234	<code>\RaggedRightParfillskip</code> . . . . .	4, 96, 119, 132, 200
<code>\newif</code> . . . . .	20, 21, 22, 139	<code>\RaggedRightParindent</code> . . . . .	4, 96, 122, 135, 201
<code>\newlength</code> . . . . .	96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109	<code>\RaggedRightRightskip</code> . . . . .	4, 96, 116, 129, 198
<code>\nobreak</code> . . . . .	160	<code>\relax</code> . . . . .	14, 17, 63, 74, 80, 91, 142, 218, 224, 230, 236
<code>\normalbaselineskip</code> . . . . .	46, 60	<code>\renewcommand</code> . . . . .	48
<code>\normallineskip</code> . . . . .	44, 59	<code>\RequirePackage</code> . . . . .	92, 94
<code>\normallineskiplimit</code> . . . . .	45, 61	<code>\reserved@e</code> . . . . .	160, 162
<b>O</b>		<code>\reserved@f</code> . . . . .	160, 162
<code>\OptionNotUsed</code> . . . . .	87	<code>\rightskip</code> . . . . .	5, 12, 17, 42, 173, 186, 199, 210
<b>P</b>		<b>S</b>	
<code>\PackageInfo</code> . . . . .	71	<code>\sloppy</code> . . . . .	47, 62
<code>\par</code> . . . . .	35, 51, 162	<code>\spaceskip</code> . . . . .	13, 143, 145, 148
<code>\parfillskip</code> . . . . .	43, 174, 187, 200, 211	<code>\string</code> . . . . .	71
<code>\parindent</code> . . . . .	7, 38, 138, 175, 188, 201, 212	<b>T</b>	
<code>\parskip</code> . . . . .	38, 54, 162	<code>\thr@@</code> . . . . .	142
<code>\PassOptionsToPackage</code> . . . . .	68, 85	<code>\trivlist</code> . . . . .	217, 223, 229, 235
<code>\ProcessOptions</code> . . . . .	91	<code>\tt</code> . . . . .	16
<b>R</b>		<code>\ttraggedright</code> . . . . .	15
<code>\RaggedLeft</code> . . . . .	3, 179, 230, 246	<code>\tw@</code> . . . . .	124, 125, 127, 129, 145
<code>\raggedleft</code> . . . . .	243, 246	<b>U</b>	
<code>\RaggedLeftLeftskip</code> . . . . .	4, 96, 112, 125, 184	<code>\unskip</code> . . . . .	158
<code>\RaggedLeftParfillskip</code> . . . . .	4, 96, 118, 131, 187	<b>X</b>	
<code>\RaggedLeftParindent</code> . . . . .	4, 96, 121, 134, 188	<code>\xspaceskip</code> . . . . .	14
<code>\RaggedLeftRightskip</code> . . . . .	4, 96, 115, 128, 185	<b>Z</b>	
<code>\RaggedRight</code> . . . . .	3, 58, 71, 72, 192, 224, 247	<code>\z@</code> . . . . .	6, 7, 12, 17, 38, 41, 57, 120, 121, 122, 124, 125, 127, 129, 133, 134, 135, 142, 143, 148, 162, 208, 209
<code>\raggedright</code> . . . . .	2, 11, 244, 247	<code>\z@skip</code> . . . . .	38, 42, 54, 113, 115, 117, 118, 126, 128, 130, 131
<code>\RaggedRightLeftskip</code> . . . . .	4, 96, 113, 126, 197		

## 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	
<code>\@raggedtwoe@abdhook</code> : New macro . . . . .	9
<code>\@raggedtwoe@document@opt</code> : New macro . . . . .	9

<code>\@raggedtwoe@everyselectfont</code> : Completely redesigned and removed <code>\RaggedSpaceskip</code> and <code>\RaggedXSpaceskip</code> .....	11
<code>\@raggedtwoe@gnewline</code> : New macro .....	12
<code>\@raggedtwoe@raggedrightboxes@opt</code> : New macro .....	7
<code>\@raggedtwoe@samed@gnewline</code> : New macro .....	12
<code>\Centering</code> : Call <code>\@raggedtwoe@everyselectfont</code> and switch <code>\gnewline</code> .....	12
<code>\JustifyingParfillskip</code> : New macro .....	10
<code>\JustifyingParindent</code> : New macro .....	10
<code>\RaggedLeft</code> : Call <code>\@raggedtwoe@everyselectfont</code> and switch <code>\gnewline</code> .....	13
<code>\RaggedRight</code> : Call <code>\@raggedtwoe@everyselectfont</code> and switch <code>\gnewline</code> .....	13
<code>\if@raggedtwoe@footmisc</code> : New macro .....	7
<code>\justifying</code> : 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 <code>raggedr</code> .....	1
Load the <code>footmisc</code> package .....	9
New command <code>\justifying</code> .....	3
New environment <code>justify</code> .....	5
New option <code>document</code> .....	6, 8
New option <code>footnotes</code> .....	6, 8
New option <code>raggedrightboxes</code> .....	6, 7
Pass all other options to <code>footmisc</code> if it's loaded .....	9
Removed <code>\RaggedSpaceskip</code> and <code>\RaggedXSpaceskip</code> .....	4, 10
Removed spaces and unneeded braces from <code>\setlength</code> ; replaced <code>plus</code> with <code>\@plus</code> .....	6
<code>justify</code> : New environment .....	14
v2.01	
<code>\@raggedtwoe@everyselectfont</code> : Removed the setting of <code>\xspaceskip</code> ..	11
v2.02	
<code>\@raggedtwoe@gnewline</code> : Bugfix: <code>\@nolerr</code> → <code>\@nolnerr</code> .....	12
General: Bugfix: <code>\if@raggedtwoe@footmisctrue</code> → <code>\@raggedtwoe@footmisctrue</code> .....	8
Removed <code>\setlength</code> .....	6
Use <code>\@flushglue</code> .....	6
<code>FlushRight</code> : Use <code>\trivlist</code> ... <code>\endtrivlist</code> instead of <code>\begin{trivlist}</code> ... <code>\end{trivlist}</code> .....	14
<code>justify</code> : Use <code>\trivlist</code> ... <code>\endtrivlist</code> instead of <code>\begin{trivlist}</code> ... <code>\end{trivlist}</code> .....	14
v2.03	
General: Bugfix: <code>footnotes</code> was actually <code>raggedrightboxes</code> .....	8
v2.04	
<code>\@raggedtwoe@abdhook</code> : Set <code>\@tocmarg</code> and use <code>\PackageInfo</code> .....	9
<code>\@raggedtwoe@everyselectfont</code> : Reset <code>\spaceskip</code> when we are not active .....	12
<code>\@raggedtwoe@raggedrightboxes@opt</code> : The setting of <code>\parindent</code> is superfluous .....	8
General: Initialize <code>\JustifyingParindent</code> with <code>\parindent</code> .....	10
Insert missing <code>\</code> .....	10
Save more commands .....	14

v2.1	
	<code>\@raggedtwoe@abdhook</code> : bugfix: Use <code>\@tocrmarg</code> only if it's defined . . . . . 9
	<code>\@raggedtwoe@gnewline</code> : Bugfix: handle math . . . . . 12
	General: bugfix: <code>\Flushleft</code> instead of <code>\FlushLeft</code> (found by Berend Hasselman) . . . . . 14
	bugfix: Load <code>footmisc</code> directly and not via <code>\AtEndOfPackage</code> (bug found by Axel Sommerfeldt) . . . . . 10
	document that document needs <code>footmisc</code> . . . . . 6
v2.2	
	<code>\@raggedtwoe@raggedrightboxes@opt</code> : Definition of <code>\@arrayparboxrestore</code> has changed . . . . . 7
	General: Change maintenance status of package . . . . . 2
	Move to git/gitlab, use <code>svninfo2</code> . . . . . 1
	Require a new version of L <sup>A</sup> T <sub>E</sub> X(2017/03/29) . . . . . 1