

The ragged2e-package*

Martin Schröder

<https://gitlab.com/TeXhackse/ragged2e>[†]

2023/04/04

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	<code>L^AT_EX</code>	2
2.2	<code>plain</code>	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	<code>originalcommands</code> option	7
6.2.2	<code>originalparameters</code> option	7
6.2.3	<code>raggedrightboxes</code> option	7
6.2.4	<code>footnotes</code> option	8
6.2.5	<code>document</code> 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 v3.5, revision #ea451c8, last revised 2023-04-04 13:49:53 +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 L ^A T _E X-commands	15

7 Acknowledgements 16

1 The problem

L^AT_EX 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 ^A T _E X document preparation system is a special version of Donald Knuth’s T _E X program. T _E X is a sophisticated program designed to produce high-quality typesetting, especially for mathematical text.” [5, p. xiii]	“The L ^A T _E X document preparation system is a special version of Donald Knuth’s T _E X program. T _E X is a sophisticated program designed to produce high-quality type-setting, especially for mathematical text.” [5, p. xiii]

2 Old “solutions”

2.1 L^AT_EX

L^AT_EX 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 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 plain

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

L^AT_EX, because L^AT_EX 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\tttraggedright{%
18   \tt
19   \rightskip\z@ plus2em\relax}
20 </plain>
```

3 Our solution

Since the `plain` solution can not be used with L^AT_EX, we have to redefine it and make it possible to configure it for personal preferences.

3.1 The macros

`\Centering` `\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.

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

3.2 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}{0pt plus 1em}
```

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

`\CenteringLeftskip` `\RaggedLeftLeftskip` `\RaggedRightLeftskip` These are the `\leftskips` 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` These are the `\rightskips` inserted by `\Centering`, `\RaggedLeft`, and
`\RaggedLeftRightskip` `\RaggedRight`.
`\RaggedRightRightskip`

“`\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` These are the `\parfillskip`s inserted by `\Centering`, `\RaggedLeft`,
`\RaggedLeftParfillskip` `\RaggedRight`, and `\justifying`.
`\RaggedRightParfillskip`
`\JustifyingParfillskip` “`\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` These are the `\parindents` used by `\Centering`, `\RaggedLeft`,
`\RaggedLeftParindent` `\RaggedRight`, and `\justifying`.
`\RaggedRightParindent`
`\JustifyingParindent` “`\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	L ^A T _E X setting	ragged2e setting [†]
<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>

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 L^AT_EX-commands `\centering`, `\raggedleft`, and `\raggedright` and the L^AT_EX-environments `center`, `flushleft`, and `flushright` remain unchanged.
It is the default.

[†]For proportional and monospaced fonts.

newcommands The L^AT_EX-commands `\centering`, `\raggedleft`, and `\raggedright` and the L^AT_EX-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^AT_EX.

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.

5 Required packages

This package requires the following packages:

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

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 `original-commands` 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.

23 \newif\if@raggedtwoe@originalparameters

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

24 \newif\if@raggedtwoe@footmisc

```

6.2 Declaration of options

6.2.1 originalcommands option

The `originalcommands` and `newcommands` options control the meaning of the L^AT_EX-commands for ragged text: If `newcommands` is used the L^AT_EX-commands are set equal to the commands defined by `ragged2e`.

```

25 \DeclareOption{OriginalCommands}{\@raggedtwoe@originalcommandstrue}
26 \DeclareOption{originalcommands}{\@raggedtwoe@originalcommandstrue}
27 \DeclareOption{NewCommands}{\@raggedtwoe@originalcommandsfalse}
28 \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`.

```

29 \DeclareOption{OriginalParameters}{\@raggedtwoe@originalparameterstrue}
30 \DeclareOption{originalparameters}{\@raggedtwoe@originalparameterstrue}
31 \DeclareOption{NewParameters}{\@raggedtwoe@originalparametersfalse}
32 \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.

```

```

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

```

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

```

34 \CheckCommand*{\@arrayparboxrestore}{%
35 \let\if@nbreak\iffalse
36 \let\if@noskipsec\iffalse
37 \let\par\@par
38 \let\-\@dischyp
39 \let\'@acci\let\'@accii\let\=@acciii

```

```

40 \parindent\z@ \parskip\z@skip
41 \everypar{}%
42 \linewidth\hsize
43 \@totalleftmargin\z@
44 \leftskip\z@skip \rightskip\z@skip \@rightskip\z@skip
45 \parfillskip\@flushglue
46 \lineskip\normallineskip
47 \lineskiplimit\normallineskiplimit
48 \baselineskip\normalbaselineskip
49 \sloppy}%

```

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

```

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

```

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

```

65 \let\@raggedtwoe@raggedrightboxes@opt\relax
66 }

```

Finally the declaration of the option.

```

67 \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.

```

68 \DeclareOption{footnotes}{%
69 \@raggedtwoe@footmisctrue
70 \PassOptionsToPackage{ragged}{footmisc}%
71 }

```

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.

```

72 \newcommand{\@raggedtwoe@abdhook}{%
73   \PackageInfo{ragged2e}{ABD: executing \string\RaggedRight}%
74   \RaggedRight
75   \@ifundefined{tocrmarg}{}{\edef\tocrmarg{\tocrmarg plus 2em}}%
76   \let\@raggedtwoe@abdhook\relax
77 }

```

`\@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.

```

78 \newcommand{\@raggedtwoe@document@opt}{%
79   \AtBeginDocument{\@raggedtwoe@abdhook}%
80   \@raggedtwoe@raggedrightboxes@opt
81   \@raggedtwoe@footmisctrue
82   \let\@raggedtwoe@document@opt\relax
83 }

```

Finally the declaration of the option.

```

84 \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.

```

85 \DeclareOption*{%
86   \if@raggedtwoe@footmisc
87     \PassOptionsToPackage{\CurrentOption}{footmisc}%
88   \else
89     \OptionNotUsed
90   \fi
91 }

```

6.3 Executing options

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

```

92 \ExecuteOptions{originalcommands,newparameters}
93 \ProcessOptions\relax

```

6.4 Loading packages

We need the `everyysel` package for older kernels.

```

94 \providecommand\IfFormatAtLeastTF{\@ifl@t@r\fmtversion}
95 \IfFormatAtLeastTF{2021/01/05}{}{\RequirePackage{everyysel}}

```

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).

```

96 \if@raggedtwoe@footmisc
97   \RequirePackage{footmisc}[2002/08/28]
98 \fi

```

6.5 Allocations

```

\CenteringLeftskip First we allocate the parameters
\RaggedLeftLeftskip 99 \newlength{\CenteringLeftskip}
\RaggedRightLeftskip 100 \newlength{\RaggedLeftLeftskip}
\CenteringRightskip 101 \newlength{\RaggedRightLeftskip}
\RaggedLeftRightskip 102 \newlength{\CenteringRightskip}
\RaggedRightRightskip 103 \newlength{\RaggedLeftRightskip}
\CenteringParfillskip 104 \newlength{\RaggedRightRightskip}
\RaggedLeftParfillskip 105 \newlength{\CenteringParfillskip}
\RaggedRightParfillskip 106 \newlength{\RaggedLeftParfillskip}
\JustifyingParfillskip 107 \newlength{\RaggedRightParfillskip}
\CenteringParindent 108 \newlength{\JustifyingParfillskip}
\RaggedLeftParindent 109 \newlength{\CenteringParindent}
\RaggedRightParindent 110 \newlength{\RaggedLeftParindent}
\JustifyingParindent 111 \newlength{\RaggedRightParindent}
112 \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.

```

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

```

```

133 \CenteringParfillskip\z@skip
134 \RaggedLeftParfillskip\z@skip
135 \RaggedRightParfillskip\@flushglue
136 \CenteringParindent\z@
137 \RaggedLeftParindent\z@
138 \RaggedRightParindent\z@
139 \fi
140 \JustifyingParfillskip\@flushglue
141 \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. $\mathrm{T}_{\mathrm{E}}\mathrm{X}$'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 `everyselectfont` [8], which allows us to define code which is (hopefully) executed after every fontchange in a $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ document.[‡]

`\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 $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ when the scope of them ends.

```

142 \newif\if@raggedtwo@spaceskip

```

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

```

143 \newcommand{\@raggedtwo@everyselectfont@active}{%
144 \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.

```

145 \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.

```

146 \spaceskip\z@
147 \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`.

```
148      \spaceskip\fontdimen\tw@\font
149      \fi
```

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

```
150  \else
151      \spaceskip\z@
152  \fi
153  }
```

If our kernel is new enough we use the kernel hook directly instead of the `everyselectfont` macro.

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

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

```
158 \let\@raggedtwoe@everyselectfont\relax
159 \if@raggedtwoe@localhook
160   \AddToHook{selectfont}[ragged2e]{\@raggedtwoe@everyselectfont}%
161 \fi
162 %      \end{macrocode}
163 % \end{macro}
164 %
165 %
166 % ^^A -----
167 %
168 % \subsection{The commands}
169 % ^^A
170 % \begin{macro}{\@raggedtwoe@savedcr}
171 % We save the definition of \cs{\textbackslash} in
172 % \cs{@raggedtwoe@savedcr}.
173 % \begin{macrocode}
174 \let\@raggedtwoe@savedcr\
```

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

```
175 \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 achieved 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.

```
176 \def\@raggedtwoe@centercrNoindent{%
177   \AddToHookNext{para/begin}{\OmitIndent}%
178   \@centercr%
179 }
```

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

```
180 \newcommand*{\@raggedtwoe@gnewline}[1]{%
181   \ifvmode
182     \@nolnerr
183   \else
184     \unskip
185     \ifmmode
```

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

```
186     \reserved@a {\reserved@f #1}\nobreak \hskip\z@ \break
187   \else
188     \reserved@a {\reserved@f #1}%
189     \@raggedtwoe@centercrNoindent
190   \fi
191 \fi
192 }
```

`\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.

```
193 \DeclareRobustCommand{\Centering}{%
194   \if@raggedtwoe@localhook
195     \let\@raggedtwoe@everyselectfont\@raggedtwoe@everyselectfont@active%
196   \fi
197   \ifx\\ \@raggedtwoe@savedcr
198     \let\\ \@raggedtwoe@centercrNoindent
199   \fi
200   \let\@gnewline\@raggedtwoe@gnewline
201   \leftskip\CenteringLeftskip
202   \@rightskip\CenteringRightskip
203   \rightskip\@rightskip
204   \parfillskip\CenteringParfillskip
205   \parindent\CenteringParindent
206   \@raggedtwoe@spaceskiptrue
207   \@raggedtwoe@everyselectfont
208 }
```

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

```
209 \DeclareRobustCommand{\RaggedLeft}{%
```

```

210 \if@raggedtwoe@localhook
211 \let\@raggedtwoe@everyselectfont\@raggedtwoe@everyselectfont@active%
212 \fi
213 \ifx\\ \@raggedtwoe@savedcr
214 \let\\ \@raggedtwoe@centercrNoindent
215 \fi
216 \let\@gnewline\@raggedtwoe@gnewline
217 \leftskip\RaggedLeftLeftskip
218 \@rightskip\RaggedLeftRightskip
219 \rightskip\@rightskip
220 \parfillskip\RaggedLeftParfillskip
221 \parindent\RaggedLeftParindent
222 \@raggedtwoe@spaceskiptrue
223 \@raggedtwoe@everyselectfont
224 }

```

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

```

225 \DeclareRobustCommand{\RaggedRight}{%
226 \if@raggedtwoe@localhook
227 \let\@raggedtwoe@everyselectfont\@raggedtwoe@everyselectfont@active%
228 \fi
229 \ifx\\ \@raggedtwoe@savedcr
230 \let\\ \@raggedtwoe@centercrNoindent
231 \fi
232 \let\@gnewline\@raggedtwoe@gnewline
233 \leftskip\RaggedRightLeftskip
234 \@rightskip\RaggedRightRightskip
235 \rightskip\@rightskip
236 \parfillskip\RaggedRightParfillskip
237 \parindent\RaggedRightParindent
238 \@raggedtwoe@spaceskiptrue
239 \@raggedtwoe@everyselectfont
240 }

```

`\justifying` `\justifying` switches back to the defaults used by L^AT_EX for typesetting justified text.

```

241 \DeclareRobustCommand{\justifying}{%
242 \if@raggedtwoe@localhook
243 \let\@raggedtwoe@everyselectfont\@raggedtwoe@everyselectfont@active%
244 \fi
245 \let\\ \@raggedtwoe@savedcr
246 \let\@gnewline\@raggedtwoe@saved@gnewline
247 \leftskip\z@
248 \@rightskip\z@
249 \rightskip\@rightskip
250 \parfillskip\JustifyingParfillskip
251 \parindent\JustifyingParindent
252 \@raggedtwoe@spaceskipfalse
253 \@raggedtwoe@everyselectfont
254 }

```

6.8 The environments

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

```

255 \newenvironment{Center}{%
256   \trivlist
257   \Centering\item\relax
258 }{%
259   \endtrivlist
260 }
261 \newenvironment{FlushLeft}{%
262   \trivlist
263   \RaggedRight\item\relax
264 }{%
265   \endtrivlist
266 }
267 \newenvironment{FlushRight}{%
268   \trivlist
269   \RaggedLeft\item\relax
270 }{%
271   \endtrivlist
272 }
```

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

```

273 \newenvironment{justify}{%
274   \trivlist
275   \justifying\item\relax
276 }{%
277   \endtrivlist
278 }
```

6.9 Overloading the L^AT_EX-commands

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

```

279 \if@raggedtwoe@originalcommands
280 \else
281   \let\LaTeXcentering\centering
282   \let\LaTeXraggedleft\raggedleft
283   \let\LaTeXraggedright\raggedright
284   \let\centering\Centering
285   \let\raggedleft\RaggedLeft
286   \let\raggedright\RaggedRight
287   \let\LaTeXcenter\center
288   \let\endLaTeXcenter\endcenter
289   \let\LaTeXflushleft\flushleft
290   \let\endLaTeXflushleft\endflushleft
291   \let\LaTeXflushright\flushright
```

```

292 \let\endLaTeXflushright\endflushright
293 \let\center\Center
294 \let\endcenter\endCenter
295 \let\flushleft\FlushLeft
296 \let\endflushleft\endFlushLeft
297 \let\flushright\FlushRight
298 \let\endflushright\endFlushRight
299 \fi

300 </package>

```

7 Acknowledgements

A first version of this package for L^AT_EX 2.09 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. `ltxmiscn.dtx`. Part of the L^AT_EX-distribution.
- [2] Robin Fairbairns. `footmisc` — a portmanteau package for customising footnotes in L^AT_EX 2_ε. <https://mirror.ctan.org/macros/latex/contrib/footmisc/>.
- [3] James Kilfinger. <https://ctan.org/tex-archive/obsolete/macros/latex/contrib/misc/raggedr.sty>. L^AT_EX 2_ε package.
- [4] Donald E. Knuth. *The T_EX Book*, volume A of *Computers & Typesetting*. Addison-Wesley, Reading, MA, USA, 1986.
- [5] Leslie Lamport. *L^AT_EX: 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 L^AT_EX 2_ε. Part of the L^AT_EX-distribution.
- [8] Martin Schröder. The obsolete everysel-package. <http://mirrors.ctan.org/macros/latex/contrib/everysel/everysel.pdf>. L^AT_EX 2_ε package.
- [9] Martin Schröder. The raggedri document option. Was in <http://mirrors.ctan.org/tex-archive/macros/latex209/contrib/raggedright>. L^AT_EX 2.09 style, outdated.
- [10] Laurent Siebenmann. ragged.sty. CTAN:tex-archive/macros/generic/ragged.sty. generic macro file for plain and L^AT_EX.

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	\@raggedtwoe@gnewline
\-	38, 54	180, 200, 216, 232
\=	39, 55	\@raggedtwoe@localhooktrue . . . 156
\@par	37, 53	\@raggedtwoe@originalcommandsfalse
\@acci	39, 55	27, 28
\@accii	39, 55	\@raggedtwoe@originalcommandstrue
\@acciii	39, 55	25, 26
\@arrayparboxrestore	34, 50	\@raggedtwoe@originalparametersfalse
\@centercr	4, 178	31, 32
\@dischyph	38, 54	\@raggedtwoe@originalparameterstrue
\@flushglue	5, 10, 45,	29, 30
114, 115, 117, 119, 122, 135, 140		\@raggedtwoe@raggedrightboxes@opt
\@gnewline . . . 175, 200, 216, 232, 246		33, 67, 80
\@ifl@t@r	94	\@raggedtwoe@samed@gnewline . .
\@ifundefined	75	175, 246
\@nolnerr	182	\@raggedtwoe@samedcr
\@plus	127, 128, 130, 132	170, 174, 197, 213, 229, 245
\@raggedtwoe@abdhook	72, 79	\@raggedtwoe@spaceskipfalse . . 252
\@raggedtwoe@centercrNoindent		\@raggedtwoe@spaceskiptrue . . .
176, 189, 198, 214, 230		206, 222, 238
\@raggedtwoe@document@opt . . 78, 84		\@rightskip 5, 6, 44, 202,
\@raggedtwoe@everyselectfont .		203, 218, 219, 234, 235, 248, 249
143, 195,		\@tocrmarg 75
207, 211, 223, 227, 239, 243, 253		\@totalleftmargin 43, 59
\@raggedtwoe@everyselectfont@active		\@ 4, 174,
143, 195, 211, 227, 243		197, 198, 213, 214, 229, 230, 245
\@raggedtwoe@footmisctrue . . 69, 81		\' 39, 55

- A**
- `\AddToHook` 160
 - `\AddToHookNext` 177
 - `\AtBeginDocument` 79
- B**
- `\baselineskip` 48, 62
 - `\begin` 170, 173
 - `\break` 186
- C**
- `\Center` 293
 - `Center (env.)` 255
 - `\center` 287, 293
 - `\Centering` 193, 257, 284
 - `\centering` 281, 284
 - `\CenteringLeftskip` 99, 114, 127, 201
 - `\CenteringParfillskip`
..... 99, 120, 133, 204
 - `\CenteringParindent`
..... 99, 123, 136, 205
 - `\CenteringRightskip`
..... 99, 117, 130, 202
 - `\CheckCommand` 34
 - `\cs` 171, 172
 - `\CurrentOption` 87
- D**
- `\DeclareOption` 25, 26, 27,
28, 29, 30, 31, 32, 67, 68, 84, 85
 - `\DeclareRobustCommand`
..... 2, 193, 209, 225, 241
 - `\def` 13, 17, 176
- E**
- `\edef` 75
 - `\else` 88, 126, 147, 150, 183, 187, 280
 - `\end` 162, 163
 - `\endCenter` 294
 - `\endcenter` 288, 294
 - `\endFlushLeft` 296
 - `\endflushleft` 290, 296
 - `\endFlushRight` 298
 - `\endflushright` 292, 298
 - `\endLaTeXcenter` 288
 - `\endLaTeXflushleft` 290
 - `\endLaTeXflushright` 292
 - `\endtrivlist` 259, 265, 271, 277
 - environments:
 - `Center` 255
 - `FlushLeft` 255
 - `FlushRight` 255
 - `justify` 273
 - `\everypar` 41, 57
- `\EverySelectfont` 157
- `\ExecuteOptions` 92
- F**
- `\fi` 90, 98, 139,
149, 152, 161, 190, 191, 196,
199, 212, 215, 228, 231, 244, 299
 - `\finalhyphendemerits` 7
 - `\FlushLeft` 295
 - `FlushLeft (env.)` 255
 - `\flushleft` 289, 295
 - `\FlushRight` 297
 - `FlushRight (env.)` 255
 - `\flushright` 291, 297
 - `\fmtversion` 94
 - `\font` 145, 148
 - `\fontdimen` 145, 148
- H**
- `\hsize` 42, 58
 - `\hskip` 186
- I**
- `\if@nobreak` 35, 51
 - `\if@noskipsec` 36, 52
 - `\if@raggedtwoe@footmisc` 24, 86, 96
 - `\if@raggedtwoe@localhook`
... 154, 159, 194, 210, 226, 242
 - `\if@raggedtwoe@originalcommands`
..... 22, 279
 - `\if@raggedtwoe@originalparameters`
..... 23, 113
 - `\if@raggedtwoe@spaceskip` . 142, 144
 - `\ifdim` 145
 - `\iffalse` 35, 36, 51, 52
 - `\IfFormatAtLeastTF` 94, 95, 155
 - `\ifmmode` 185
 - `\ifvmode` 181
 - `\ifx` 197, 213, 229
 - `\item` 257, 263, 269, 275
- J**
- `justify (env.)` 273
 - `\justifying` 241, 275
 - `\JustifyingParfillskip` 99, 140, 250
 - `\JustifyingParindent` . 99, 141, 251
- L**
- `\LaTeXcenter` 287
 - `\LaTeXcentering` 281
 - `\LaTeXflushleft` 289
 - `\LaTeXflushright` 291
 - `\LaTeXraggedleft` 282
 - `\LaTeXraggedright` 283
 - `\leftskip` .. 8, 44, 201, 217, 233, 247

- `\let` 4,
35, 36, 37, 38, 39, 51, 52, 53,
54, 55, 65, 76, 82, 158, 174,
175, 195, 198, 200, 211, 214,
216, 227, 230, 232, 243, 245,
246, 281, 282, 283, 284, 285,
286, 287, 288, 289, 290, 291,
292, 293, 294, 295, 296, 297, 298
- `\lineskip` 46, 61
- `\lineskiplimit` 47, 63
- `\linewidth` 42, 58
- N**
- `\newcommand` 33, 72, 78, 143, 180
- `\newenvironment` . 255, 261, 267, 273
- `\newif` 22, 23, 24, 142, 154
- `\newlength` 99,
100, 101, 102, 103, 104, 105,
106, 107, 108, 109, 110, 111, 112
- `\nobreak` 186
- `\normalbaselineskip` 48, 62
- `\normallineskip` 46, 61
- `\normallineskiplimit` 47, 63
- O**
- `\OmitIndent` 177
- `\OptionNotUsed` 89
- P**
- `\PackageInfo` 73
- `\par` 37, 53
- `\parfillskip` . 45, 204, 220, 236, 250
- `\parindent`
.. 9, 40, 141, 205, 221, 237, 251
- `\parskip` 40, 56
- `\PassOptionsToPackage` 70, 87
- `\ProcessOptions` 93
- `\providecommand` 94
- R**
- `\RaggedLeft` 209, 269, 285
- `\raggedleft` 282, 285
- `\RaggedLeftLeftskip`
..... 99, 115, 128, 217
- `\RaggedLeftParfillskip`
..... 99, 121, 134, 220
- `\RaggedLeftParindent`
..... 99, 124, 137, 221
- `\RaggedLeftRightskip`
..... 99, 118, 131, 218
- `\RaggedRight` 60, 73, 74, 225, 263, 286
- `\raggedright` 3, 13, 283, 286
- `\RaggedRightLeftskip`
..... 99, 116, 129, 233
- `\RaggedRightParfillskip`
..... 99, 122, 135, 236
- `\RaggedRightParindent`
..... 99, 125, 138, 237
- `\RaggedRightRightskip`
..... 99, 119, 132, 234
- `\relax` 16, 19, 65, 76, 82,
93, 145, 158, 257, 263, 269, 275
- `\renewcommand` 50
- `\RequirePackage` 95, 97
- `\reserved@e` 186, 188
- `\reserved@f` 186, 188
- `\rightskip`
6, 14, 19, 44, 203, 219, 235, 249
- S**
- `\sloppy` 49, 64
- `\spaceskip` 15, 146, 148, 151
- `\string` 73
- `\subsection` 168
- T**
- `\textbackslash` 171
- `\thr@@` 145
- `\trivlist` 256, 262, 268, 274
- `\tt` 18
- `\ttraggedright` 17
- `\tw@` 127, 128, 130, 132, 148
- U**
- `\unskip` 184
- 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, 186, 247, 248
- `\z@skip` 8, 40, 44, 56, 116,
118, 120, 121, 129, 131, 133, 134

Change History

v1.00	
General: New from <code>raggedright</code> 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 <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>	3, 10
Removed spaces and unneeded braces from <code>\setlength</code> ; replaced <code>plus</code> with <code>\@plus</code>	6
<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	13
<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>	13
<code>\if@raggedtwoe@footmisc</code> : New macro	7
<code>justify</code> : New environment	15
<code>\justifying</code> : New macro	14
<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>	14
v2.01	
<code>\@raggedtwoe@everyselectfont</code> : Removed the setting of <code>\xspaceskip</code>	11
v2.02	
General: Bugfix: <code>\if@raggedtwoe@footmisctrue</code> → <code>\@raggedtwoe@footmisctrue</code>	8
Removed <code>\setlength</code>	6
Use <code>\@flushglue</code>	6
<code>\@raggedtwoe@gnewline</code> : Bugfix: <code>\@nolerr</code> → <code>\@nolnerr</code>	13
<code>FlushRight</code> : Use <code>\trivlist</code> ... <code>\endtrivlist</code> instead of <code>\begin{trivlist}</code> ... <code>\end{trivlist}</code>	15
<code>justify</code> : Use <code>\trivlist</code> ... <code>\endtrivlist</code> instead of <code>\begin{trivlist}</code> ... <code>\end{trivlist}</code>	15

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

\justifying: Bugfix: remove spurious space	14
\RaggedLeft: Bugfix: remove spurious space	13
\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	13
\RaggedRight: Bugfix: Only activate the global hook	14